

UNIVERSIDADE FEDERAL DO PARANÁ

JOSUÉ OLIVEIRA CAMARGO  
MAILSON GONÇALVES CORRÊA

DIÁRIO DE CLASSE E CIA: UM SISTEMA DE ORGANIZAÇÃO DAS ATIVIDADES  
DOCENTES

CURITIBA,  
2015

UNIVERSIDADE FEDERAL DO PARANÁ

JOSUÉ OLIVEIRA CAMARGO  
MAILSON GONÇALVES CORRÊA

DIÁRIO DE CLASSE E CIA: UM SISTEMA DE ORGANIZAÇÃO DAS ATIVIDADES  
DOCENTES

Trabalho de conclusão de curso  
apresentado como requisito parcial à  
conclusão do curso de Tecnologia em  
Análise e Desenvolvimento de Sistemas.  
Setor de Educação Profissional e  
Tecnológica da Universidade Federal do  
Paraná.

Orientação: Prof. Paulo Vitor dos Santos  
Zeferino

CURITIBA,  
2015

## SUMÁRIO

RESUMO.....	4
ABSTRACT.....	5
LISTA DE FIGURAS .....	6
1. INTRODUÇÃO .....	8
1.1 A EDUCAÇÃO .....	8
1.2 OBJETIVOS .....	10
1.2.1 OBJETIVO GERAL .....	10
1.2.2 OBJETIVOS ESPECÍFICOS .....	10
1.3 ORGANIZAÇÃO DO TEXTO .....	11
2. MARCO TEÓRICO.....	12
3. MATERIAL E MÉTODOS.....	14
4. APRESENTAÇÃO DO SISTEMA .....	20
5. TESTES.....	49
6. CONSIDERAÇÕES FINAIS .....	65
AGRADECIMENTOS .....	66
REFERÊNCIAS.....	66
APÊNDICE A – ESTRUTURA ANALÍTICA DO PROJETO (EAP) .....	69
APÊNDICE B – GRÁFICO DE GANTT.....	70
APÊNDICE C – DIAGRAMAS DE SEQUÊNCIA .....	71
APÊNDICE D – DIAGRAMA DE CLASSES .....	75
APÊNDICE E- DIAGRAMA ENTIDADE RELACIONAMENTO (DER) .....	76

## RESUMO

Ao longo do tempo, a educação sofreu transformações graduais, e desde 1950 a tecnologia digital auxilia o processo de ensino-aprendizagem. As LMS's são ferramentas didáticas de apoio que vem se tornando cada vez mais usadas hoje em dia. Entretanto, o sistema educacional não acompanhou a revolução tecnológica, e continuou utilizando as mesmas metodologias tradicionais. Um professor leva cerca de vinte por cento da aula para fazer a chamada, o que se considera “tempo perdido” (Shinkawa, 2014). O presente trabalho desenvolve um *software* para auxiliar o professor nos processos de organização docente, tais como o controle de notas e frequência, entre outras funcionalidades de interesse docente, visando uma otimização no tempo dos profissionais da educação. Essa ferramenta foi desenvolvida utilizando o processo metodológico RUP de desenvolvimento de software, modelado com UML 2 e codificado predominantemente em linguagem C# voltado para web, e utilizando o *framework* de desenvolvimento gráfico para web *Bootstrap*, já que a interface do *software* deve ser desenvolvida de maneira a evitar a rejeição do usuário (De Moura Junior, ). O sistema de diário de classe virtual é uma alternativa para aqueles que desejam inovar nos métodos de organização docente, pois oferece ao usuário um meio robusto de organização profissional.

**Palavras-chave:** Inovação Tecnológica, Professor, *Software*, Gerenciamento de Sala de Aula, *Learning Management Systems*.

## **ABSTRACT**

*Over time, education has undergone gradual transformations, and since 1950 digital technology aids the teaching-learning process. The LMS's are teaching tools that support is becoming increasingly used today. However, the educational system did not follow the technological revolution, and continued using the same traditional methods. A teacher takes about twenty percent of the class to make the call, which is considered "lost time" (Shinkawa, 2014). This paper develops a software to help the teacher in teaching organizational processes, such as control of notes and frequency, among other features of teaching interest, aimed at optimizing the time of education professionals. This tool was developed using the methodological process software development RUP, modeling with UML 2 and encoded predominantly in C # facing website, and using the graphical development framework for web Bootstrap, since the software interface should be developed so to avoid user rejection (De Moura Junior). The virtual class diary system is an alternative for those wishing to innovate in teaching methods of organization as it offers the user a robust means of professional organization.*

**Keywords:** *Innovation, Teacher, Software, Classroom Management, Learning Management Systems.*

## LISTA DE FIGURAS

Figura 1: Plano de Riscos.....	11
Figura 2: Diagrama de casos de uso.....	13
Figura 3: Tela de login.....	16
Figura 4: Tela modal de auto cadastro de professores.....	17
Figura 5: Tela inicial.....	18
Figura 6: Tela de CRUD de turma.....	19
Figura 7: Tela modal de cadastro de turma.....	20
Figura 8: Tela de CRUD de aluno.....	21
Figura 9: Tela de horário das aulas.....	22
Figura 10: Tela modal de cadastro de aula no horário.....	23
Figura 11: Tela de CRUD de chamada.....	24
Figura 12: Tela de modal de cadastro de nova chamada.....	25
Figura 13: Tela de modal de chamada.....	26
Figura 14: Tela de chamada rapida.....	27
Figura 15: Tela de CRUD de Tema.....	28
Figura 16: Tela de CRUD de questões.....	29
Figura 17: Tela de modal para cadastrar questão.....	30
Figura 18: Tela de CRUD de prova.....	31

Figura 19: Tela de modal de notas.....	32
Figura 20: Tela de geração de prova.....	33
Figura 21: Tela de Prova gerada.....	34
Figura 22: Tela de CRUD de Trabalho.....	35
Figura 23: Tela de modal de notas do trabalho.....	36
Figura 24: Tela de relatório de presença.....	37
Figura 25: Tela de relatório de notas.....	38
Figura 26: Tela de Inicial do administrador.....	39
Figura 27: Tela de modal de cadastro de professores para administrador.....	40
Figura 28: Tela de CRUD de Instituição.....	41
Figura 29: Tela de modal de cadastro de instituição.....	42
Figura 30: Tela de CRUD de matéria.....	43
Figura 31: Tela de relatório administrativo.....	44

## **1. INTRODUÇÃO**

### **1.1 A EDUCAÇÃO**

Para Freire (1987), a educação sofreu, ao longo do tempo, com um sistema “bancário” de ensino, em que a tarefa básica do professor era “depositar o conhecimento” no aluno e posteriormente “sacar” esse conhecimento em forma de uma prova. Tal sistema demonstrava-se inconsistente desde o início, a começar com os jargões escolares: Os alunos (literalmente os “sem luz”) eram treinados a decorar e repetir tudo aquilo o que o professor com um ar de superioridade “professava”. Dessa forma, os alunos deixariam a instituição de ensino formados (ou seja, no “formato desejado”). Embora ainda exista educadores com esse pensamento, em geral a educação tomou novos caminhos e começou a ser vista de uma maneira mais humanizada, na qual o aluno deixa de ser um simples “repetidor” passivo, passando a ser mais autônomo, formulando e promovendo parte significativa dos seus conhecimentos (D’Ignázio, 1992).

O pensamento que se tinha sobre o papel da escola evoluiu. Para Freire (2000), “Ensinar não é transferir conhecimentos, mas criar possibilidades para sua produção ou sua construção”. Desse modo, a escola passou a buscar maneiras de facilitar a produção de conhecimento, fato que anteriormente acontecia apenas em ambientes de ensino superior.

Com o advento tecnológico, a informática aparece para auxiliar o processo educacional em meados de 1950, quando os primeiros computadores começam a ser produzidos. A introdução da informática na educação era uma tentativa de



implementação da máquina de ensinar idealizada por Skinner, e tinha a função de “memorizar” uma sequência de conteúdos e repassa-los para o aluno, as chamadas CAI’s - Instruções auxiliadas por computador (Valente, 1997).

Neste contexto, a tecnologia computacional deu um salto, e todos os recursos didáticos passaram a utilizar essas ferramentas. Para Mussoi et al. (2010), “[...] a tecnologia pode ser usada como instrumento de apoio para enriquecer as aulas inovando as atividades didático-pedagógicas”, ou seja, a informática passa a ser usada no processo de ensino-aprendizagem, já que permite que professores e estudantes possam interagir de maneira mais significativa com os conteúdos a serem estudados, ou mesmo como uma forma de organização. Nesse contexto, surgiram os objetos de aprendizagem (OA), que foram definidos por Tauroco et al. (2004) como “materiais educacionais com objetivos pedagógicos que servem para apoiar o processo de ensino-aprendizagem”.

Entre esses objetos de aprendizagem, existem aqueles que não entrarão em contato direto com aluno. Tratam-se de ferramentas computacionais que auxiliam o professor na organização da sua vida profissional. Os LMS’s (do inglês, *Learning Management Systems*) ou Sistemas de Gestão da Aprendizagem, são *softwares* que possibilitam a análise e o processamento do conteúdo informativo, bem como a geração de relatórios e diagramas que facilitam a condução e aperfeiçoamento do processo de ensino ensino-aprendizagem (Lobato,2008).

Lévy (1999, p 173) destaca a importância do uso da tecnologia como uma forma de atualizar as práticas pedagógicas:

Como manter as práticas pedagógicas atualizadas com esses novos processos de transação de conhecimento? Não se trata aqui de usar as tecnologias a qualquer custo, mas sim de *acompanhar consciente e deliberadamente uma mudança de civilização* que questiona profundamente

as formas institucionais, as mentalidades e a cultura dos sistemas educacionais tradicionais e sobretudo os papéis de professor e de aluno.

O sistema educacional encontra-se em um estado de sobrecarga. A busca dos indivíduos por conhecimento nunca foi tão intensa quanto é atualmente, porém as metodologias não serão capazes de suprir essa demanda. Para esse mesmo autor, não é possível formar professores na mesma proporção em que a sociedade anseia por saber. Para solucionar esse grande empecilho, o autor também relata que é necessário tomar medidas que ampliem a capacidade pedagógica do professor, de forma que as novas demandas estudantis encontrem formas alternativas - e ao mesmo tempo atrativas - de se aprender.

## **1.2 OBJETIVOS**

A seguir, os objetivos norteadores deste trabalho.

### **1.2.1 OBJETIVO GERAL**

Desenvolver um sistema que permita e auxilie o controle dos documentos que compõe o diário de classe dos professores, tais como registro de frequência e de atividades, de tal forma a otimizar o tempo do profissional da educação.

### **1.2.2 OBJETIVOS ESPECÍFICOS**

- Desenvolver para o sistema, uma interface amigável aos usuários (professores);

- Realizar a modelagem deste utilizando a UML (Linguagem de Modelagem Unificada);
- Aprimorar os conhecimentos relacionados ao uso do framework Bootstrap

### **1.3 ORGANIZAÇÃO DO TEXTO**

Nos próximos capítulos, dá-se a compreensão da organização desse trabalho.

No capítulo 2 (Marco Teórico) são apresentados alguns conceitos referentes à educação, bem como a metodologia de funcionamento e a importância do diário de classe para o bom andamento do processo de ensino-aprendizagem. O capítulo 3 (Material e Métodos) nos dá uma visão panorâmica de como o sistema foi modelado, bem como os recursos e ferramentas usadas para tal, como a EAP (Estrutura Analítica do Projeto) e a UML (Linguagem de Modelagem Unificada), Plano de Atividades e Gráfico de Gantt, as responsabilidades da equipe e os planos de risco e custo do projeto, além de estar descrito o processo de elaboração e concepção do Diário desta aplicação web. No capítulo 4 (apresentação do software) encontra-se o tutorial de instalação e manipulação do referido. O capítulo 5 (testes) contém a validação que o sistema sofreu – tanto caixa branca quanto caixa preta. Já o capítulo 6 (considerações finais) encontramos algumas perspectivas futuras sobre a incrementação e manutenção do software desenvolvido. Nas Referências consta toda a base informacional utilizada para desenvolver esse trabalho. Nos Apêndices, toda a modelagem (Tabelas e diagramas): Diagrama de casos de uso e suas especificações, DER(Diagrama entidade-relacionamento), Diagrama de classes, sequencia, etc...

## 2. MARCO TEÓRICO

O diário de classe desempenha um papel essencial na sala de aula.

(...) via de regra, chegou à escola como um caderno de chamada, um caderno de registro de frequência de alunos e alunas, de datas das aulas e conteúdos desenvolvidos a cada uma dessas aulas, reduzindo este a cumprir um papel bastante, e talvez unicamente, burocrático por aqueles que dele faziam uso (Pecoits, 2009, p. 31).

Conforme a normativa do IFRS (Instituto Federal de Educação, Ciência e Tecnologia do Rio Grande do Sul), são componentes do diário de classe:

- a. O registro de frequência;
- b. O registro de conteúdo, estratégias de aprendizagem e atividades desenvolvidas;
- c. Relatório de notas;
- d. Tabela de notas parciais;
- e. Ata de Estudos Orientados e;
- g. Anotações de recuperação paralela – na qual devem ser descritas atividades realizadas especificamente com os estudantes que apresentarem necessidade de recuperação específica de conteúdos.

Por se tratar de um documento oficial, que pode ser requisitado pela justiça se necessário e por conter um caráter de preenchimento compulsório por parte do docente (GS/SEED, 2012), o diário de classe não admite rasuras, o que exige do professor um grande estresse no que diz respeito ao preenchimento, dado à

responsabilidade que está envolvida nesse processo. Desta forma, se houver algum erro, este deverá ser refeito, em muitas vezes, devido ao estilo de impressão, ocorre quase que em todos os casos na íntegra do período em questão (bimestre, trimestre). Entretanto, esse tempo poderia ser aproveitado na preparação de aulas ou no desenvolvimento do professor pedagógico do profissional da educação (Shinkawa, 2014).

De acordo com Merli & Stangherlim (2013), o diário de classe é um documento que deve ser reflexivo, analisador de práticas, um repensador da ação pedagógica. Entretanto, vem sofrendo sérias críticas em seu estado atual. Com a constante inovação na sociedade tecnológica, os alunos estão cada vez mais dinâmicos e o professor deve ao mesmo tempo que leva nas costas o fardo burocrático das notas e frequências, manter os alunos motivados, preparando boas aulas ou ao menos passando seu tempo em sala de aula lecionando. Levando em consideração que em média leva-se dez minutos para se realizar a chamada em cada aula, e sabendo que normalmente, um período comporta cinco aulas de cinquenta minutos, o professor deixa de lecionar uma aula\dia para realizar a chamada (Shinkawa, 2014). Isso corresponderia a vinte por cento do total da carga horária de um docente. Se desconsiderarmos a hora-atividade, um professor poderia perder em média um dia de trabalho semanal com o instrumento.

Outro aspecto é, conforme Tragtenberg (1985) que a presença obrigatória do "Diário de Classe", nas mãos do professor, anotando faltas e comparecimentos nuns casos, arbitrando "meia falta" ao aluno que atrasou uns minutos ou saiu mais cedo da aula, é uma técnica de controle pedagógico burocrático por excelência e gênero herdada do presídio.

### **3. MATERIAL E MÉTODOS**

Após o recolhimento de informações literárias, foi desenvolvido uma aplicação para computador. O sistema propôs informatizar o burocrático processo do diário de classe, substituindo o uso do papel pelo sistema, otimizando desta maneira o tempo dos professores. Esse software fará a organização pessoal dos dados profissionais de cada docente, individualmente. Isso proporciona uma facilitação dos processos burocráticos, além de servir como parte integrante e indispensável do processo de ensino-aprendizagem, uma vez que substituirá definitivamente os registros físicos. É proposto também um banco de questões para que os professores possam montar trabalhos, provas e atividades de maneira mais ágil. A ferramenta computacional foi codificada nas linguagens de programação C# (Mukihi, 2002), HTML (Raggatt, Dave et al., 1999), CSS (Collison, 2006) e JavaScript (Flanagan, 2004) voltados para Web.

A C# ou C Sharp (em português lê-se "cê charp"), é uma linguagem de programação orientada a objetos desenvolvida pela Microsoft como parte da plataforma .NET. A sua sintaxe orientada a objetos foi baseada no C++, entretanto inclui outras influências de linguagens de programação, como Object Pascal e Java (Microsoft, 2014).

#### **3.1 PLANO DE RISCOS**

Para o sistema, foi adotado o plano de riscos a seguir:

	Risco	Data Limite	Consequencia	Ação	Monitoramento	Probabilidade	Impacto	Classificação
1	Mudanças no escopo do projeto (Imprecisão dos requisitos)	-	Atrasos no cronograma de desenvolvimento	Validação dos requisitos junto ao cliente, criação de lista de melhorias	Avaliar a mudança. Caso seja imprescindível, calcular atraso. Se a mudança for melhoria, colocar na lista de melhorias se não acarretar maiores impactos.	Alta	Alto	6
2	Imprecisão de Prazos	-	Atrasos no cronograma de desenvolvimento	Elaborar cronograma e distribuir as atividades do projeto conforme a disponibilidade e capacidade técnica de cada um	Monitorar as atividades e fazer "multirão" para acelerar as atividades em atraso no projeto	Alta	Alto	6
3	Indisponibilidade dos membros da equipe.	-	Atrasos no cronograma de desenvolvimento	Utilização de Redes Sociais para comunicação	Envio de mensagens online diárias para comunicar aos membros da equipe qual atividade se está realizando.	Media	Medio	3
4	Incapacidade técnica da equipe	08/04/2015	Atrasos no cronograma de desenvolvimento ou Inviabilidade do projeto	Optar por ferramentas de uso cotidiano ou de fácil aprendizagem.	Avaliar a capacidade técnica da equipe e se caso necessário, prover treinamento	Baixa	Alto	5
5	Plano de testes deficitado	02/05/2015	Atrasos no cronograma de desenvolvimento	Gerar uma documentação eficiente e Desenvolver casos de teste baseados na Especificação de Casos de Uso	Execução dos casos de teste para a validação das regras de negócio implementadas	Media	Medio	3
6	Mudança de tecnologia utilizada	04/04/2015	Atrasos no cronograma de desenvolvimento	Utilização de tecnologias a equipe domine	Verificação de domínio da tecnologia por parte da equipe	Baixa	Medio	2

Figura 1. Plano de Riscos

### **3.2 METODOLOGIA DE DESENVOLVIMENTO**

O sistema foi modelado com base na UML 2.0 (Linguagem de Modelagem Unificada), que é uma linguagem que nos fornece todos os atributos conceituais e gráficos para a modelagem esquemática de software. Com base na Orientação à Objetos, tem influência internacional na área de engenharia de Software. Guedes (2009) descreve o Diagrama de Casos de Uso como “essencial”. Segundo a notação, quem de alguma forma se comunica com o sistema é chamado de ator, e é representado por um boneco. O ator é a representação de qualquer meio externo de comunicação com o sistema, como por exemplo um usuário ou uma impressora. As funcionalidades do sistema são representadas por elipses e são chamadas de casos de uso são as tarefas que o sistema pode realizar. Algumas tarefas são agrupadas para facilitar a representação gráfica. O grupo de tarefas são similares e normalmente dizem respeito a manipulação de um objeto dentro do sistema. O Diagrama de Casos de Uso (Figura 2), traz os casos de uso Manter Questão, Manter Instituição, Manter Professores, Manter Atividade e Manter Turma. Tais casos de uso agrupam as funções de cadastro, edição e exclusão dos dados referidos em seu nome. A interação entre atores e os casos de uso são representas por linhas, e nada mais representam se não quais funcionalidades interagem externamente.



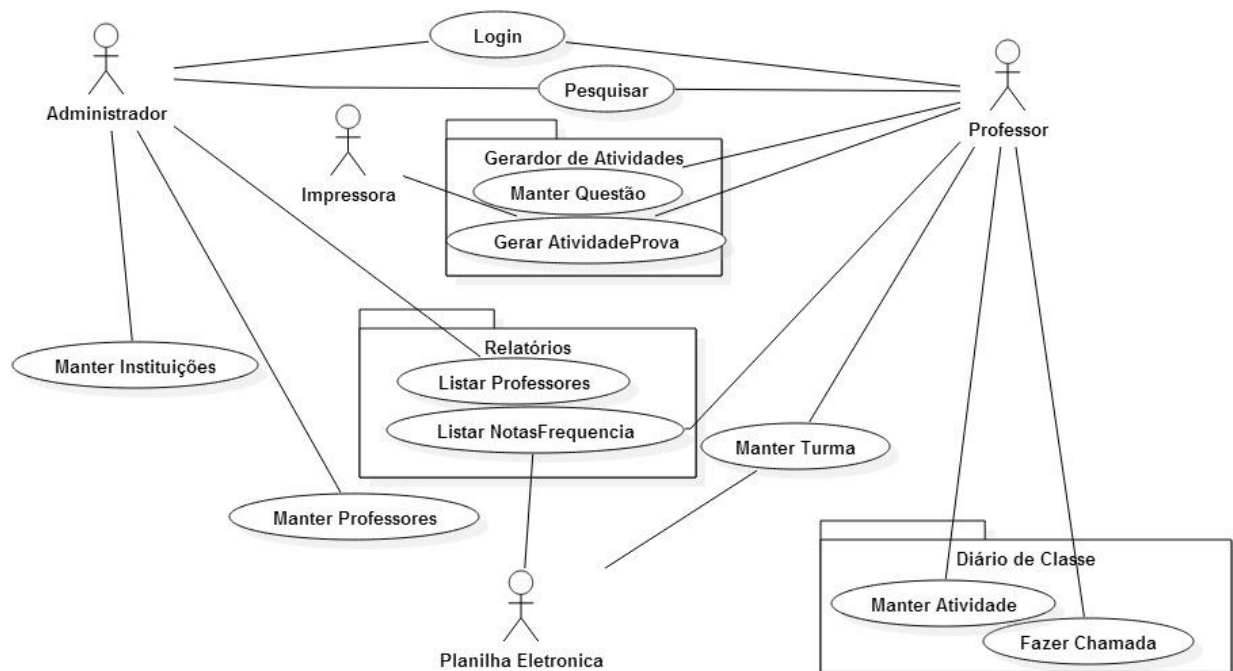


Figura 2. Diagrama de Casos de Uso.

O processo de engenharia de software escolhido para nortear o desenvolvimento do sistema foi o R.U.P (Processo Unificado da Rational), o qual é composto pela aplicação da UML e pela fragmentação do desenvolvimento em iterações incrementais, bem como organizando tarefas e responsabilidades referentes ao desenvolvimento do utilitário de maneira sistemática (Kruchten, 2000). Para representar os processos do RUP em conjunto com as especificações do sistema, consequência do levantamento de requisitos, foi necessário formalizar a WBS (*Work breakdown structure*) ou EAP (Estrutura Analítica do Projeto), que dividiu a construção do sistema em cinco etapas:

**Iniciação** -> Subdivide-se em Modelagem de Negócios (Sessão necessária para se obter entendimento do negócio) e Requisitos (No qual é feita a especificação dos casos de uso, que fornece uma visão mais técnica baseada no negócio e voltada para o software).

**Elaboração** -> Subdivide-se em iteração I e II. No primeiro ciclo, encontramos a etapa "Requisitos", que dá origem ao Diagrama de Casos de Uso (Figura 2) e ao protótipo das interfaces, o que se entende como a modelagem inicial do software. Conta ainda com a etapa de "análise e Design", contendo os modelos de análise e design do sistema a ser construído. A segunda iteração é composta pelo desenvolvimento das interfaces, construção dos demais diagramas UML e criação do Banco de Dados (que são parte da etapa "Análise e Design"). E, iniciando a codificação do software em subsistemas, temos a etapa de "Implementação".

**Construção** -> Apresenta iteração única, que se divide em "Implementação", etapa em que são codificados para uma linguagem de programação todos os casos de uso descritos e "Testes", quando é o sistema sofre avaliações para aceitação por parte do usuário final, bem como para a correção de possíveis bugs ou processos instáveis.

**Transição** -> Subdivide-se em Implantação, em que ocorre a geração dos artefatos de instalação e documentação, além do treinamento do usuário que utilizará o sistema. Aqui, Gerenciamento de Configuração e Mudança faz nova auditoria.

**Gêrencia** -> Fase que percorre todo o desenvolvimento do projeto. Subdivide-se em Planejamento e Acompanhamento do Projeto (um elabora o Plano de Riscos, o outro apresenta os Releases, relatórios e realiza reuniões periódicas). Há ainda o Encerramento do Projeto, etapa em que se conclui o projeto de desenvolvimento da aplicação.

A EAP está disponível no anexo A deste trabalho, bem como o Diagrama de GANTT.

Para que a realização do presente trabalho fosse possível, foram usados dois notebooks e um PC (*Personal Computer*) e tanto esse como aqueles possuíam processador I7 com memória RAM de 4GB cada um. Além dos computadores, uma vasta gama de ferramentas e softwares de desenvolvimento foram utilizados, entre eles:

- MS Office Word v.2013 para a redação teórica do trabalho;
- MS Office Excel v. 2013 para os cálculos (estimativa de tempo);
- MS Project v. 2013 para a elaboração de gráficos (WBS/GANTT);
- MS Visual Studio v. 2013 Community Express para a codificação do sistema;
- MS SQL Server v. 2014 Express para hospedagem de banco de dados;
- MS Team Foundation Server 2012 Express para controle de versão;
- StarUML v. 2.0.2.0 para a elaboração da documentação UML do sistema;
- Bootstrap v. 3.3.5 para a elaboração do layout das interfaces.

## 4. APRESENTAÇÃO DO SISTEMA

A seguir, as principais telas do sistema de diário de classe virtual:

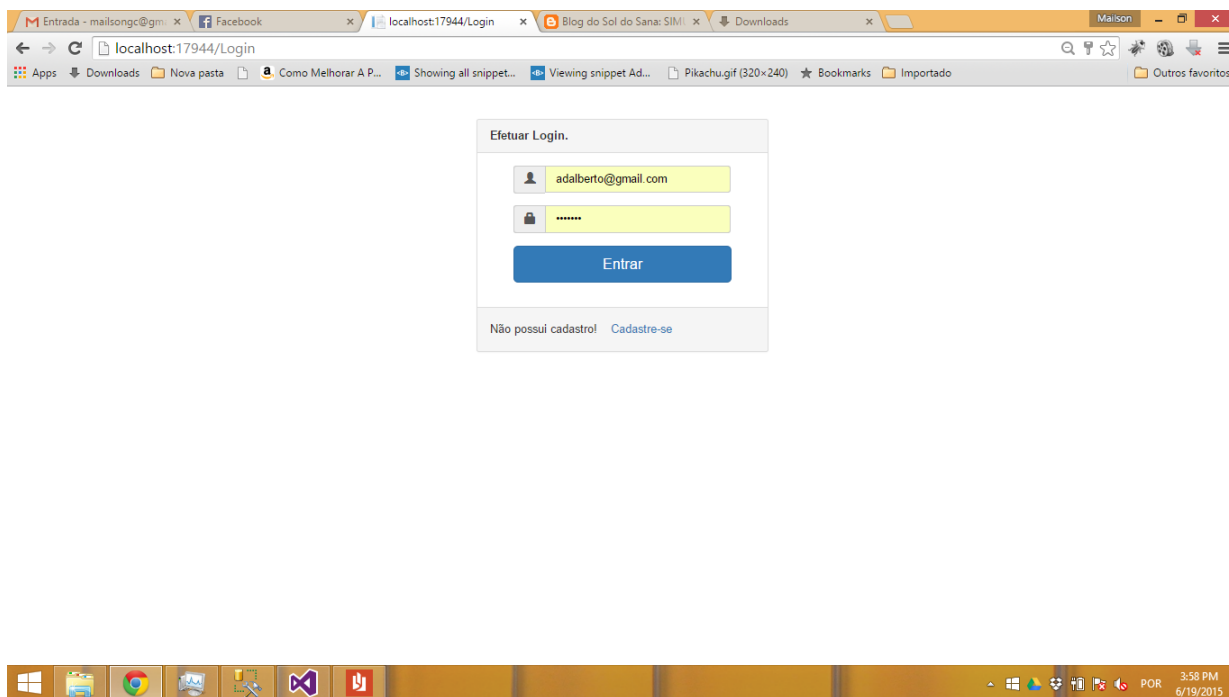


Figura 3. Tela de Login.

Tela de login: para acessar o diário de classe online, o professor deve possuir um nome de usuário e senha. Após preenchidos os campos, clica-se no botão ENTRAR (figura 3). Caso o professor ainda não possua cadastro, ao clicar em CADASTRE-SE, o usuário é redirecionado para a tela de cadastro, e pode obter as credenciais para logar no sistema.

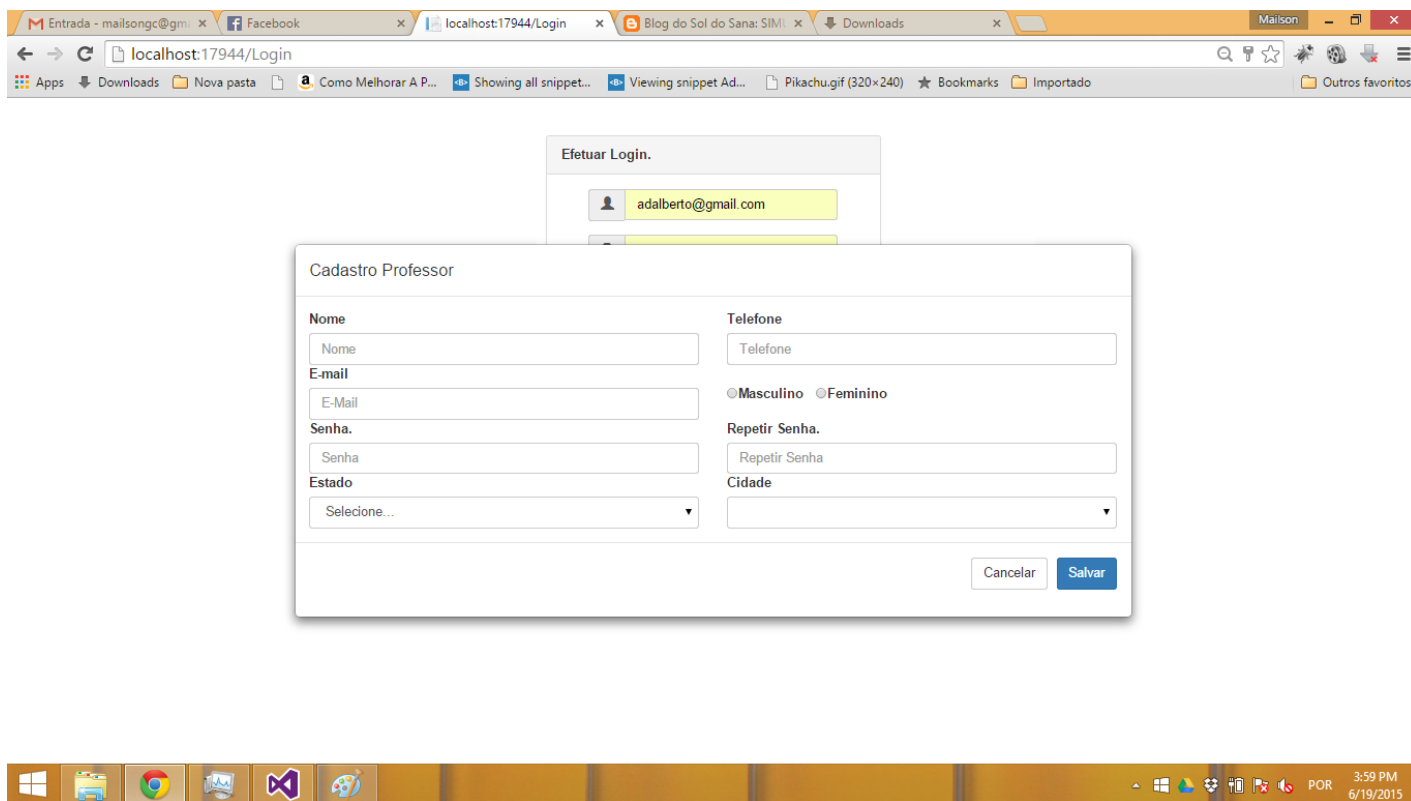


Figura 4. Tela modal de auto cadastro de professores.

Para efetuar o cadastro no sistema, o usuário deve preencher os campos da tela de cadastro de docentes, que consiste em NOME, TELEFONE, SEXO, EMAIL, SENHA, ESTADO e CIDADE conforme a figura 4. Para finalizar o cadastro, clica-se em SALVAR e o sistema exibe uma mensagem de êxito, redirecionando para a tela de login para que se possa acessar a aplicação.

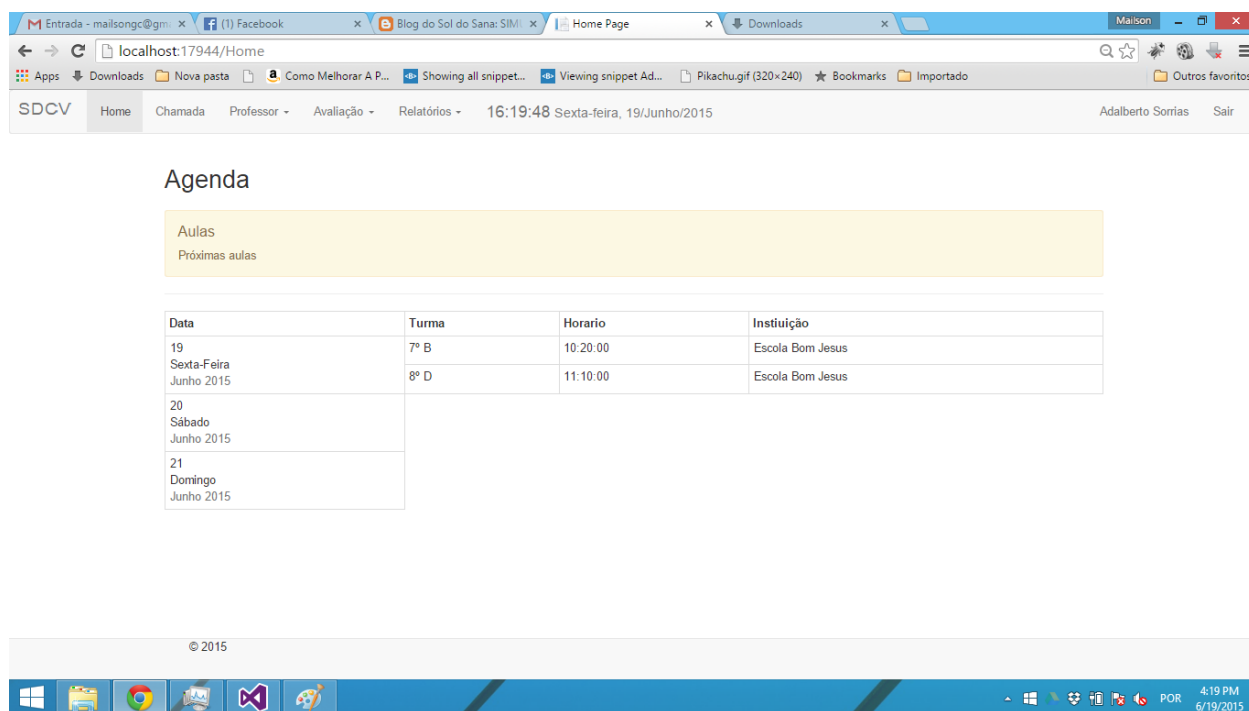


Figura 5. Tela Inicial.

Na tela inicial, são mostradas todas as aulas futuras do professor usuário do sistema nos próximos três dias. Essa tela forma uma espécie de “agenda do professor”, para que não seja necessário que o professor vá até a tela de horário para ter acesso às próximas aulas. Na tela é disponível para o professor a DATA das próximas aulas, bem como TURMA, INSTITUIÇÃO E HORÁRIO da aula. Ressalta-se que todas telas com exceção do login possuem um link para efetuar logoff como mostra a figura 5.

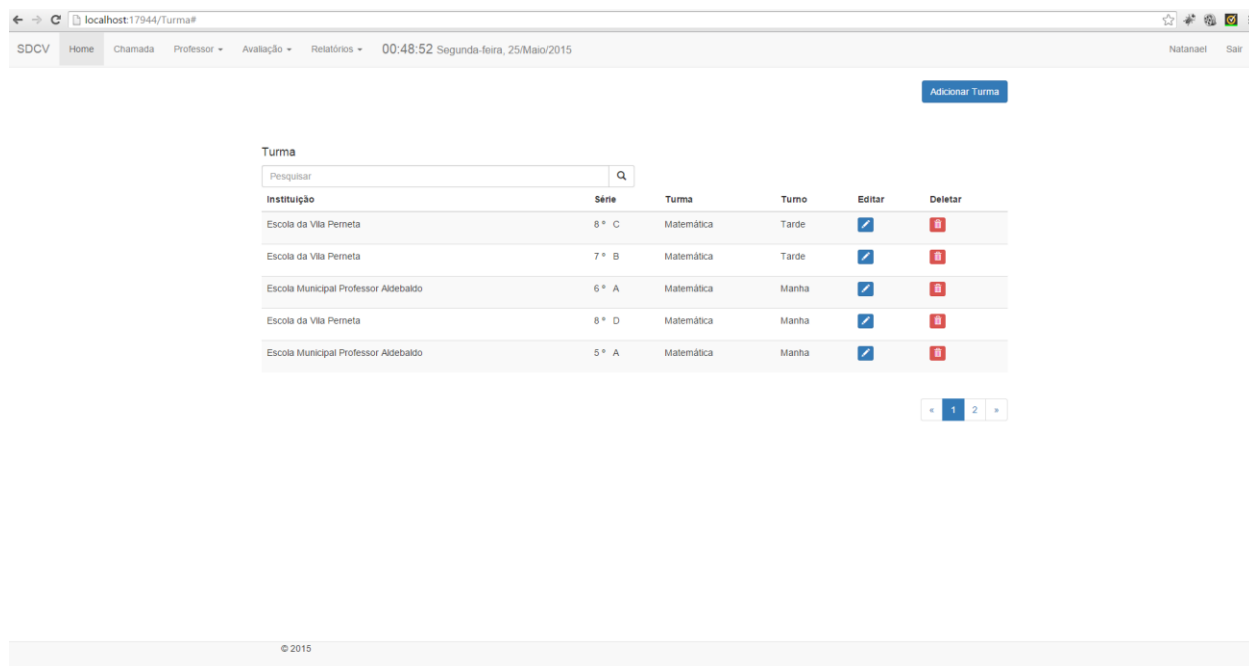


Figura 6. Tela de CRUD de turma.

Na tela de CRUD de Turma, o docente gerencia as turmas com as quais possui algum vínculo letivo durante o ano. Nesta tela é possível pesquisar as turmas por instituição, série, turma ou turno. É nesta tela também encontramos os botões que redireciona para eventuais edições ou exclusões, além inserir uma nova turma, de acordo com a figura 6.

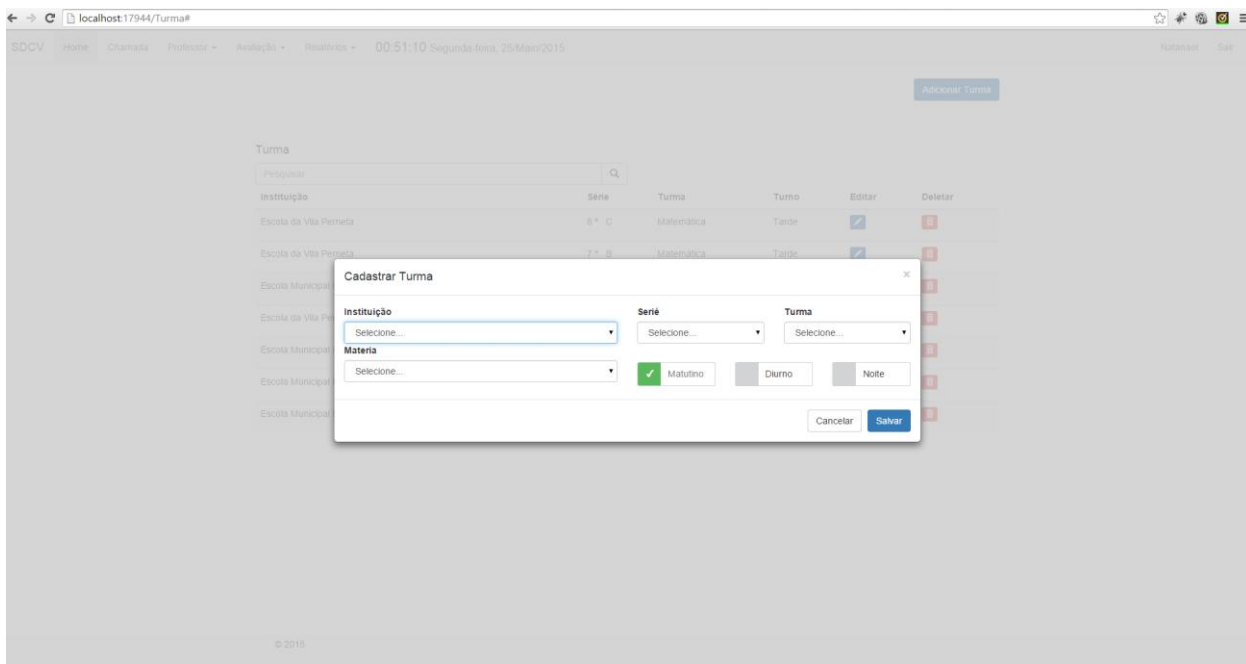


Figura 7. Tela modal de Cadastro de Turma.

Ao selecionar o botão Adicionar Turma no CRUD, o usuário é redirecionado para a tela de cadastro de turmas (figura 7). Para cadastrar uma nova turma é necessário preencher os campos INSTITUIÇÃO, MATÉRIA, SÉRIE, TURMA e TURNO, e clicar na opção SALVAR para enviar os dados para serem computados no sistema. Após o cadastro, a turma fica disponível para receber horário de aula, alunos (posteriormente cadastrados), e todos os demais processos que envolvem a turma. Se clicado em Editar, no CRUD, a mesma tela de cadastro é usada para a edição de uma turma, carregando os dados da turma a ser editada. Para concluir, salva-se normalmente a operação.



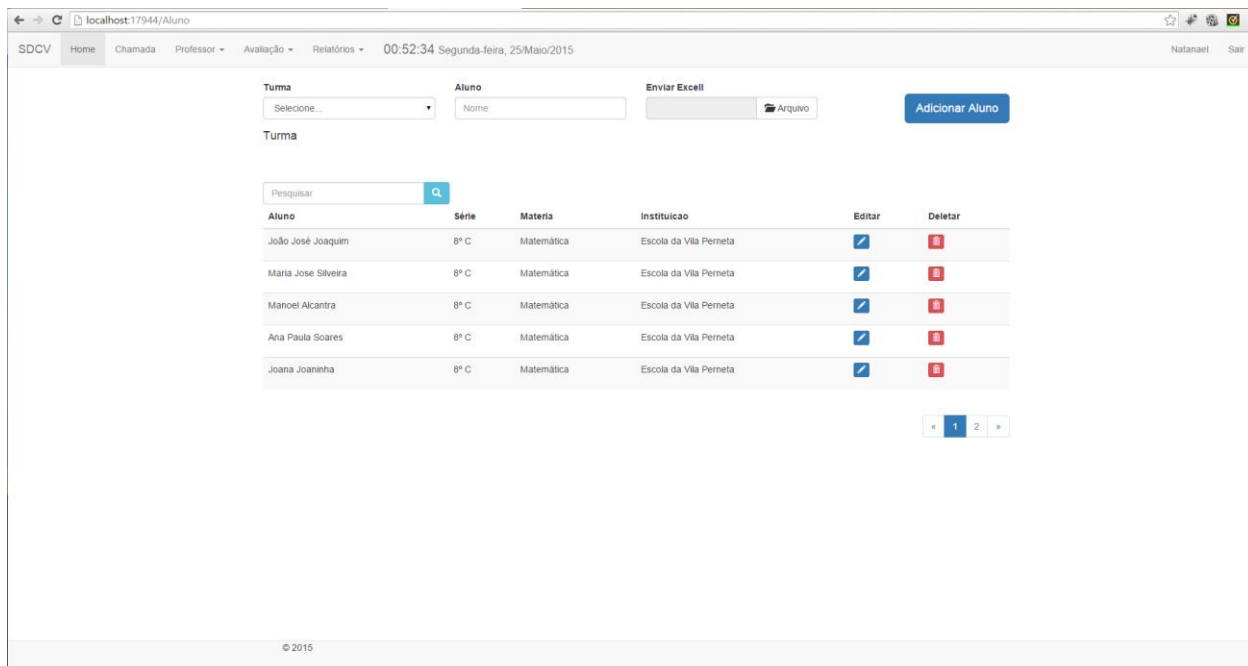


Figura 8. CRUD de aluno.

Ao ser redirecionado para esta tela, o usuário pode pesquisar para editar ou DELETAR os alunos de uma determinada TURMA, que deve apontada no campo referido. Para adicionar alunos nesta, o professor tem duas alternativas: manualmente, ao clicar em ADICIONAR ALUNO, e preenchendo NOME e TURMA a qual ele pertence, ou enviando um arquivo de planilha eletrônica (\*.xls, \*.xlsx, \*.ods, \*.xml) contendo os nomes dos alunos ao clicar em ARQUIVO do campo enviar excell (figura 8).

Aulas

Hora	Segunda-Feira	Terça-Feira	Quarta-Feira	Quinta-Feira	Sexta-Feira
07:30					
08:20		8º C Matemática			
09:10	8º C Matemática		7º B Matemática		
10:00	Intervalo	Intervalo	Intervalo	Intervalo	Intervalo
10:20				8º C Matemática	
11:10			7º B Matemática	8º C Matemática	8º D Matemática
13:30					
14:20					
15:10					
16:00	Intervalo	Intervalo	Intervalo	Intervalo	Intervalo
16:20					
17:10					

© 2015

Figura 9. Tela de horário das aulas.

Na tela de horário, o professor pode montar seu horário para uma melhor organização do seu tempo. Na tela de visualização de horário (Figura 9), pode-se adicionar uma aula clicando na célula que corresponde ao dia e horário desta, para preencher os dados da tela de cadastro de aulas (Figura 10). Após a inserção, a aula fica visível e o horário não poderá mais ser usado, a menos que aquela aula seja removida.

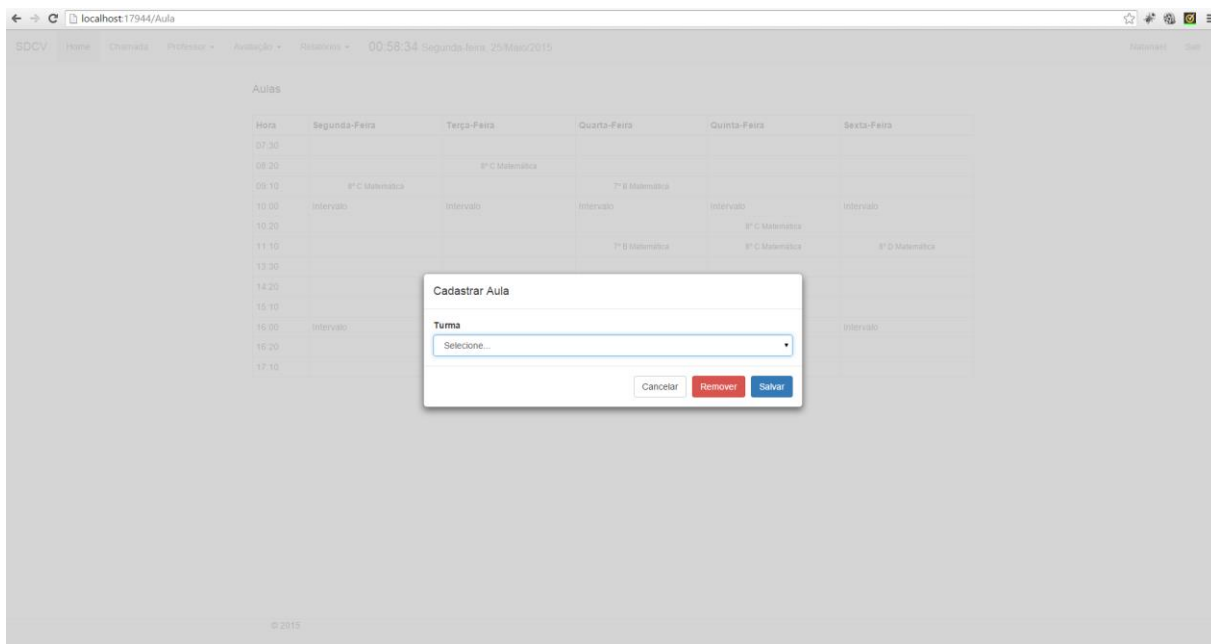


Figura 10. Tela modal de cadastro de aula no horário

Nesta tela, o docente pode selecionar a TURMA a qual a aula selecionada no horário será ministrada. Após o preenchimento, o botão SALVAR insere a aula no horário do professor. Ainda na tela de cadastro de aulas, é possível remover a aula do horário clicando no botão REMOVER como ilustra a figura 10.

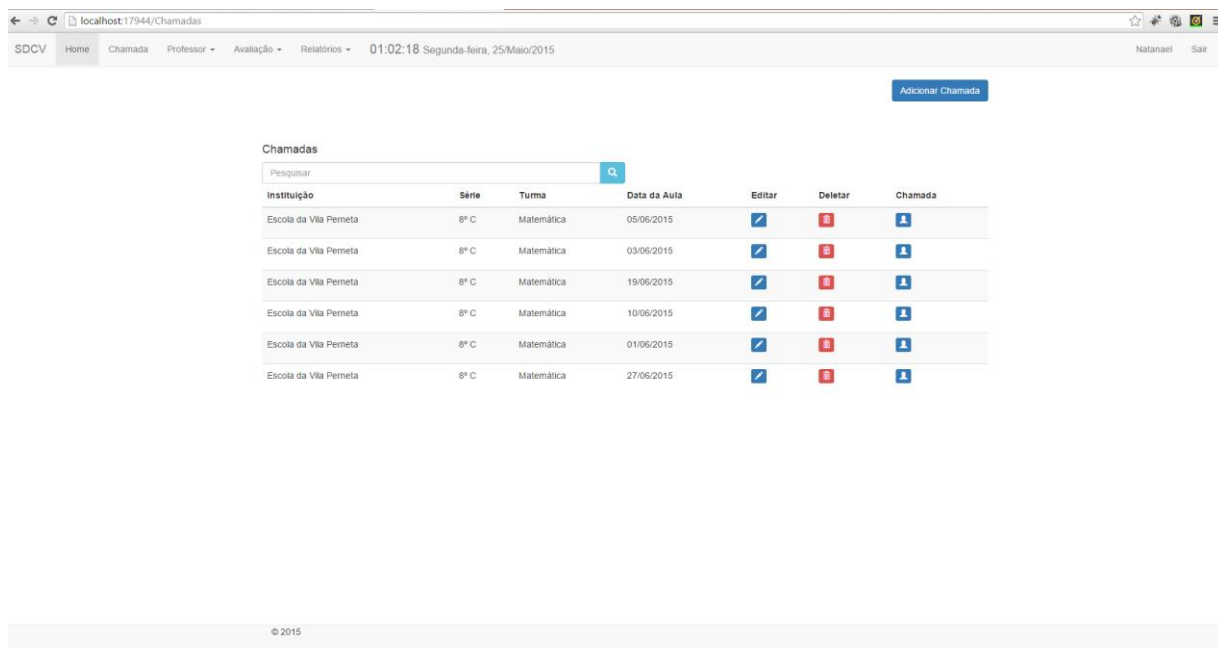


Figura 11. CRUD de chamada.

O CRUD de chamada permite ao professor pesquisar as chamadas cadastradas no sistema por DATA, TURMA, DISCIPLINA E INSTITUIÇÃO, e, além de das funções EDITAR ou DELETAR uma chamada pesquisada (figura 11), ao clicar em ADICIONAR CHAMADA é aberto o modal de cadastro para uma nova chamada (Figura 12). Ao clicar no ícone CHAMADA, abre-se um modal para inserir presença aos alunos da turma(Figura 13).

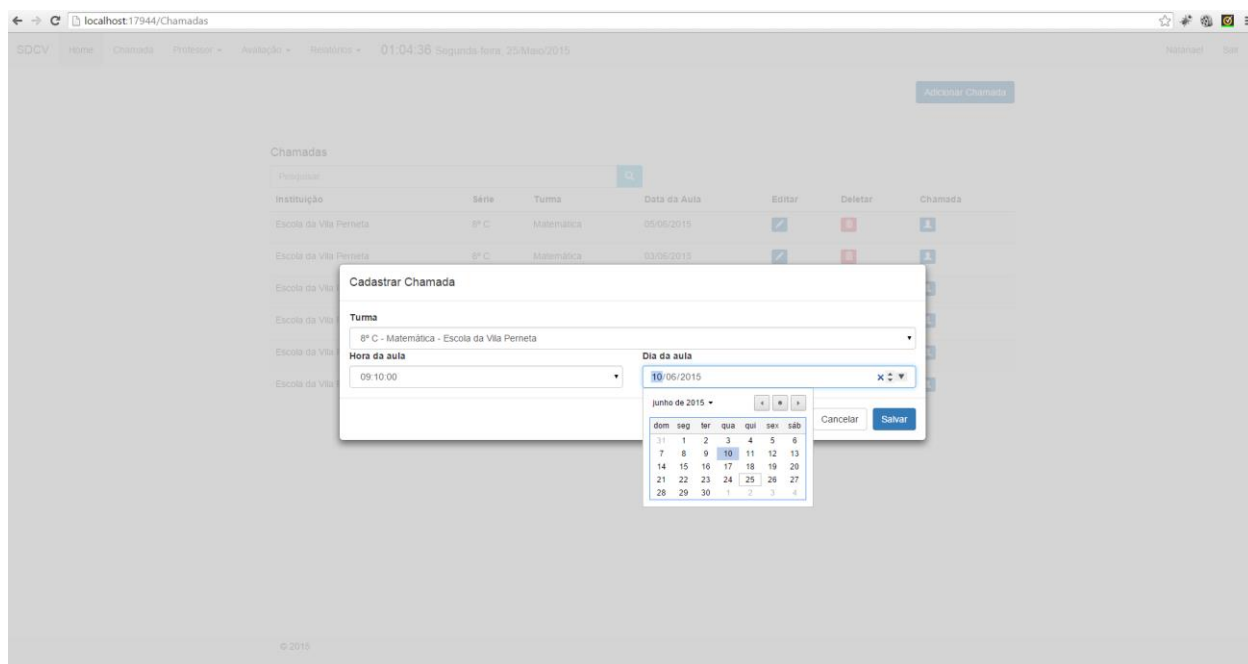


Figura 12. Tela modal de cadastro de nova chamada.

Conforme a figura 12, ao clicar em Adicionar chamada no CRUD, o professor tem acesso ao modal de cadastro. Após preencher os campos TURMA, HORA DA AULA e DATA DA AULA, basta clicar em SALVAR para que a nova chamada seja cadastrada. É possível ainda cancelar a ação clicando em CANCELAR.

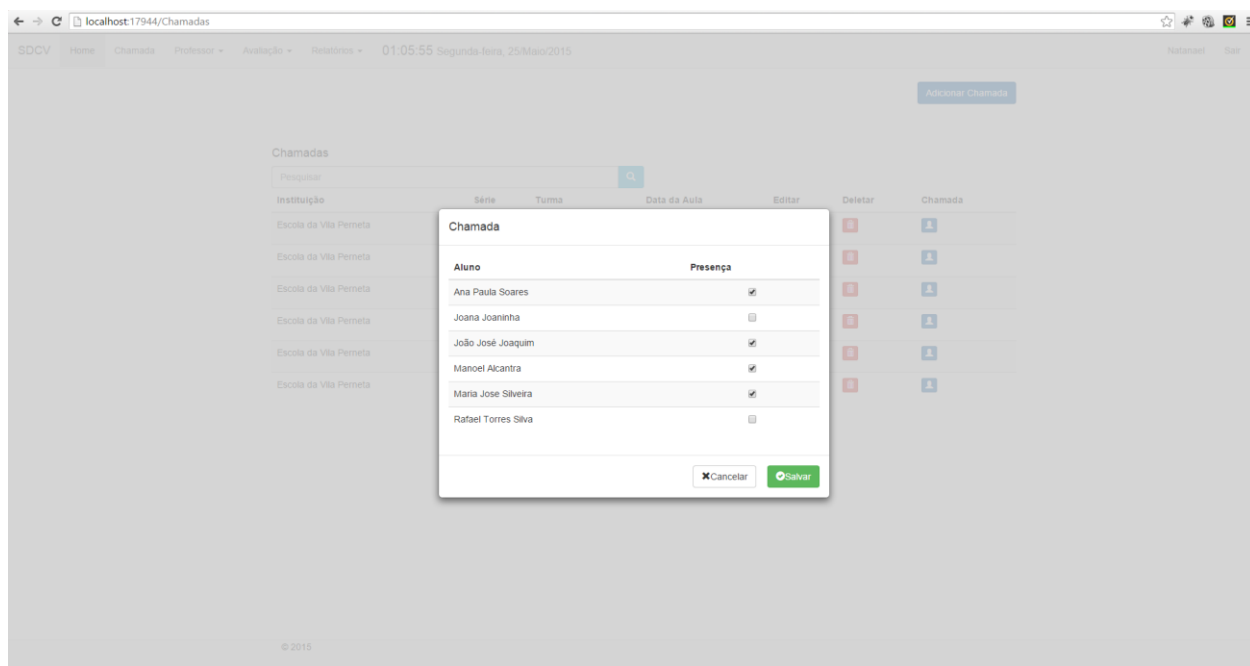


Figura 13. Tela modal de chamada.

No modal da figura 13, o usuário atribui presença aos alunos da turma marcando o checkbox da coluna PRESENÇA na linha do aluno em questão. Para que os dados possam ser guardados, clica-se em SALVAR. É possível ainda cancelar a ação, através do botão CANCELAR.

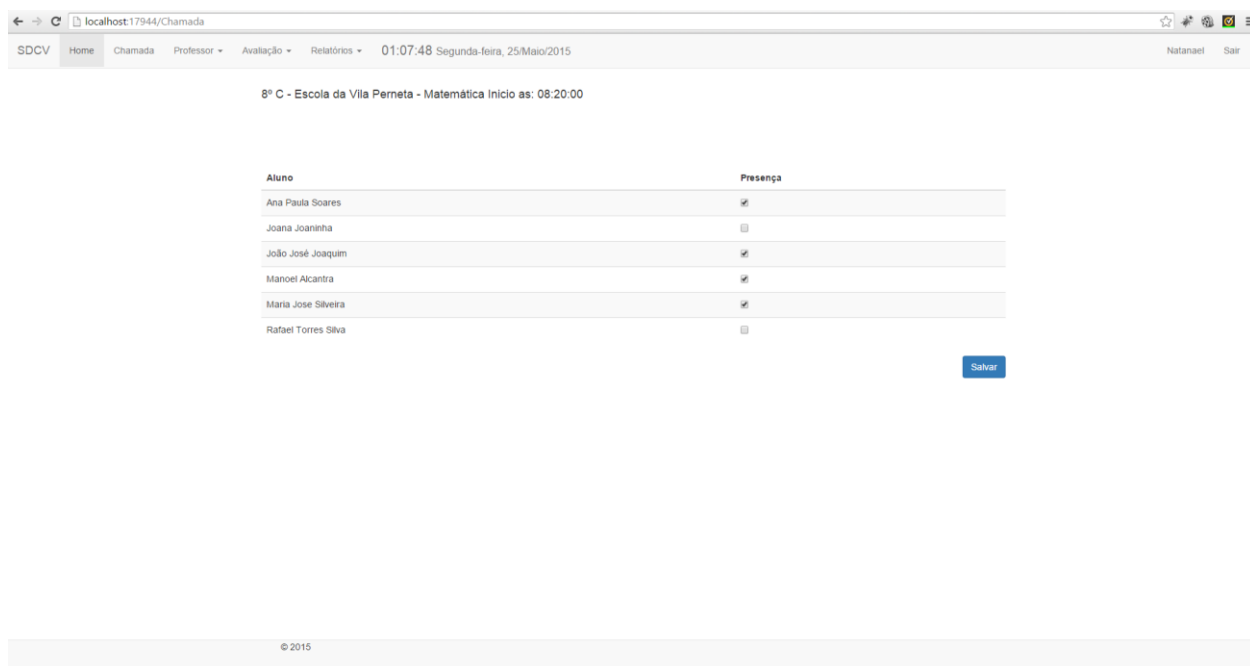


Figura 14. Tela de chamada rápida.

A figura 14 indica a tela de chamada rápida. Esta tela é acessada pela opção CHAMADA do menu superior. Nela, é mostrada a chamada da aula que acontece no momento em que a página foi aberta. Se não houver aula cadastrada para o horário atual, a pagina acessa a chamada da próxima aula.

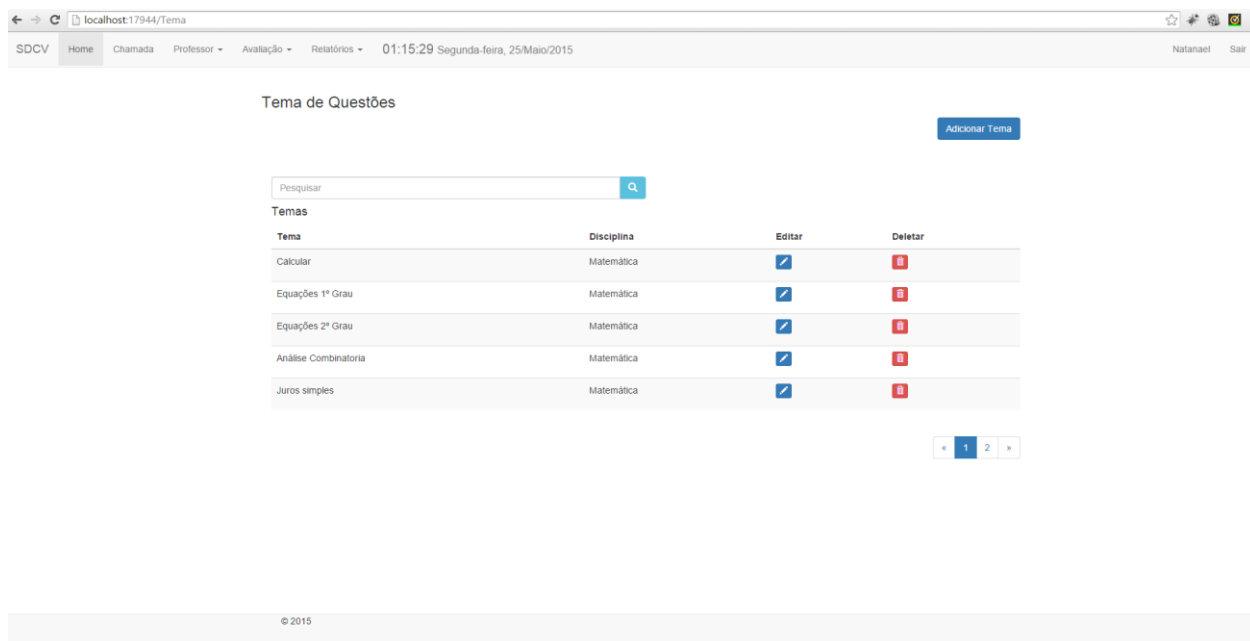


Figura 15. CRUD de Tema.

No CRUD de tema, tela ilustrada pela figura 15, o docente pode pesquisar os temas questões por TEMA e DISCIPLINA, além de EDITAR e DELETAR os temas pesquisados. Além disso, o professor, ao clicar em ADICIONAR TEMA, tem acesso a um modal similar os demais de cadastro, em que se pode inserir e salvar um novo tema para as questões.



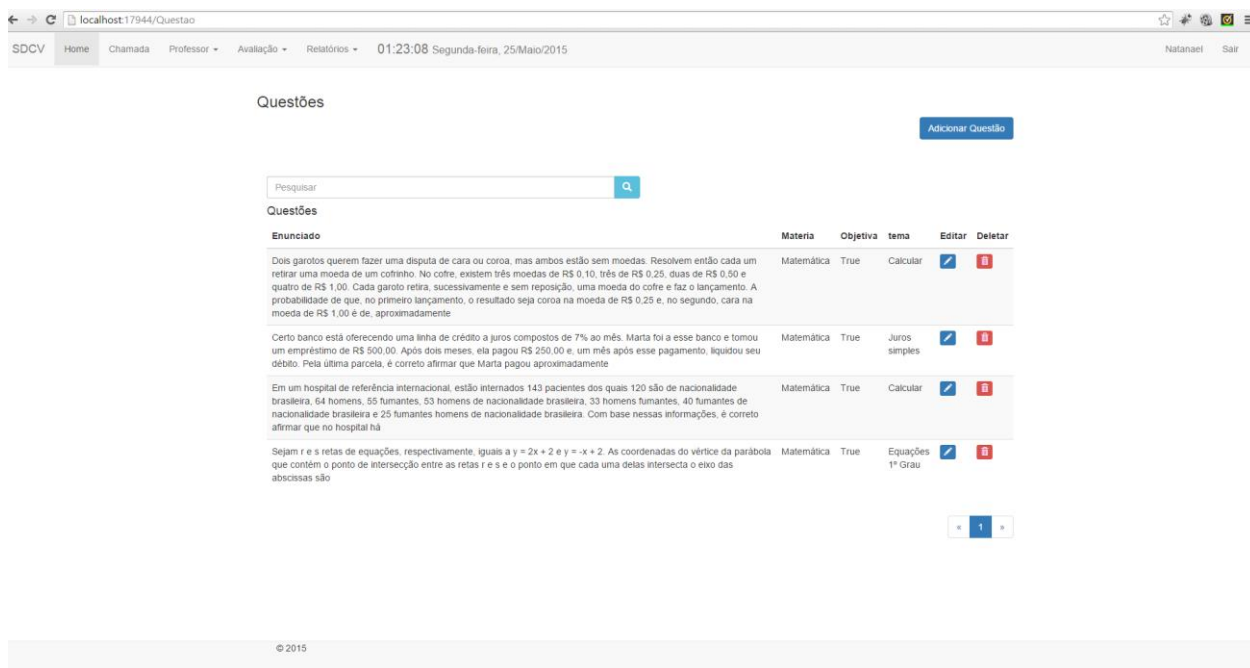


Figura 16. CRUD de questões.

Neste CRUD de questões, o professor pode pesquisar as questões cadastradas no sistema de diário de classe virtual por partes do ENUNCIADO, MATÉRIA ou por TEMA. As operações permitidas na tela são EDITAR ou DELETAR questão. Ao clicar em ADICIONAR QUESTÃO, abre-se uma modal para a inserção de nova questão no sistema (Figura 16).

The image shows a web browser window with the URL 'localhost:17944/Questao'. The page title is 'Questões'. A modal window titled 'Cadastro de Questão' is open in the center. The modal contains the following fields:

- Enunciado**: A large text area for the question text.
- Matéria**: A dropdown menu with 'Selecione...' as the placeholder.
- Tema**: A dropdown menu with 'Selecione...' as the placeholder.
- Dificuldade**: A dropdown menu with 'Selecione...' as the placeholder.
- Objetiva**: A checkbox labeled 'Objetiva'.
- Alternativa A**: A text area for the first alternative.
- Alternativa B**: A text area for the second alternative.
- Alternativa C**: A text area for the third alternative.
- Alternativa D**: A text area for the fourth alternative.
- Alternativa E**: A text area for the fifth alternative.
- Resposta**: A dropdown menu with 'Selecione...' as the placeholder.

At the bottom right of the modal are two buttons: 'Cancelar' and 'Salvar'. The background of the page shows a list of questions and a sidebar with navigation links like 'SDCV', 'Home', 'Chamada', 'Professor', 'Avaliação', 'Respostas', and a clock showing '01:25:21 Segunda-feira, 25/Março/2015'.

Figura 17. Tela modal para cadastrar questão.

O modal da figura 17 é acessado quando o professor deseja inserir uma nova questão no sistema. Para isso, basta preencher o ENUNCIADO da questão, a MATÉRIA a qual ela pertence, qual o seu TEMA, a DIFICULDADE (fácil, médio, difícil) e as ALTERNATIVAS A, B, C, D, E caso seja uma questão OBJETIVA. É necessário ainda apontar qual seria a RESPOSTA da questão. Para finalizar, o botão SALVAR cadastra a questão no sistema, deixando-a disponível para ser utilizada. Para abortar a operação, clica-se no botão CANCELAR.

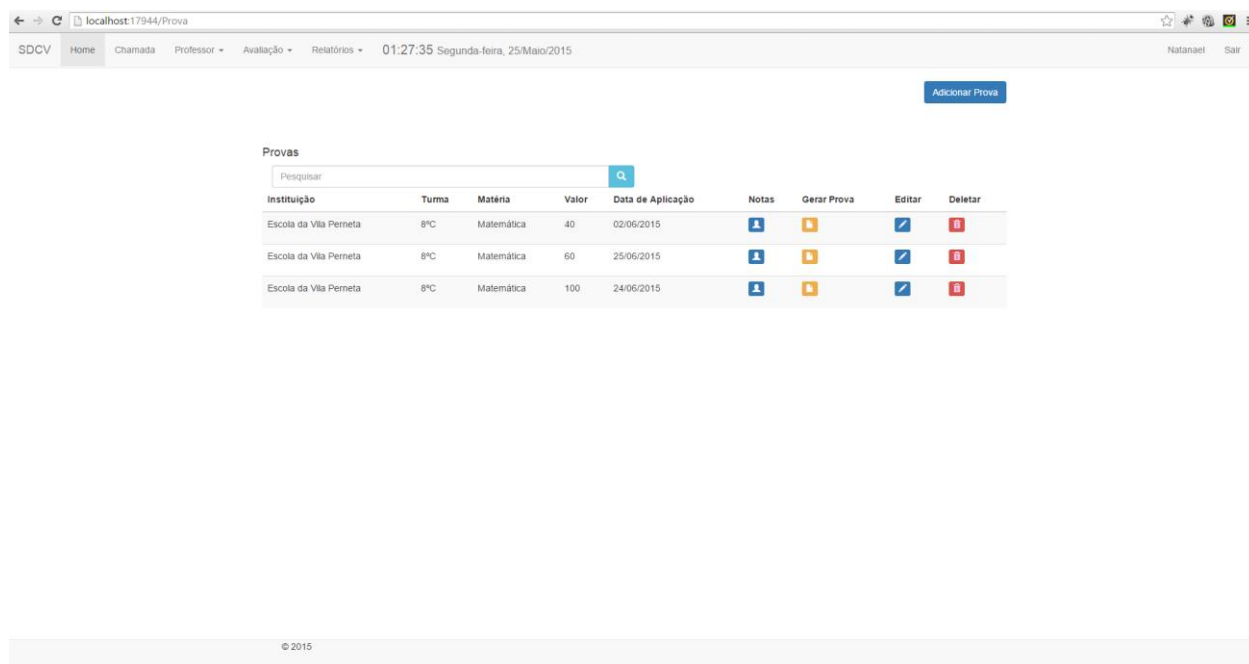


Figura 18. CRUD de prova.

Neste CRUD de prova, o educador pode pesquisar as provas cadastradas no sistema por partes do INSTITUIÇÃO, MATÉRIA, TURMA, DATA DE APLICAÇÃO ou por VALOR. É permitido EDITAR ou DELETAR a prova pesquisada conforme a figura 18. Ao clicar em ADICIONAR PROVA, tem acesso a um modal similar os demais de cadastro, em que se pode inserir e salvar uma nova prova. Ao clicar no ícone da coluna NOTAS para a prova pesquisada, é permitido ao professor, por meio de um modal, atribuir a nota aos alunos referente a prova (Figura 19). Ainda é possível ao professor criar uma prova por meio do sistema, clicando no ícone da coluna GERAR PROVA (Figura 20).

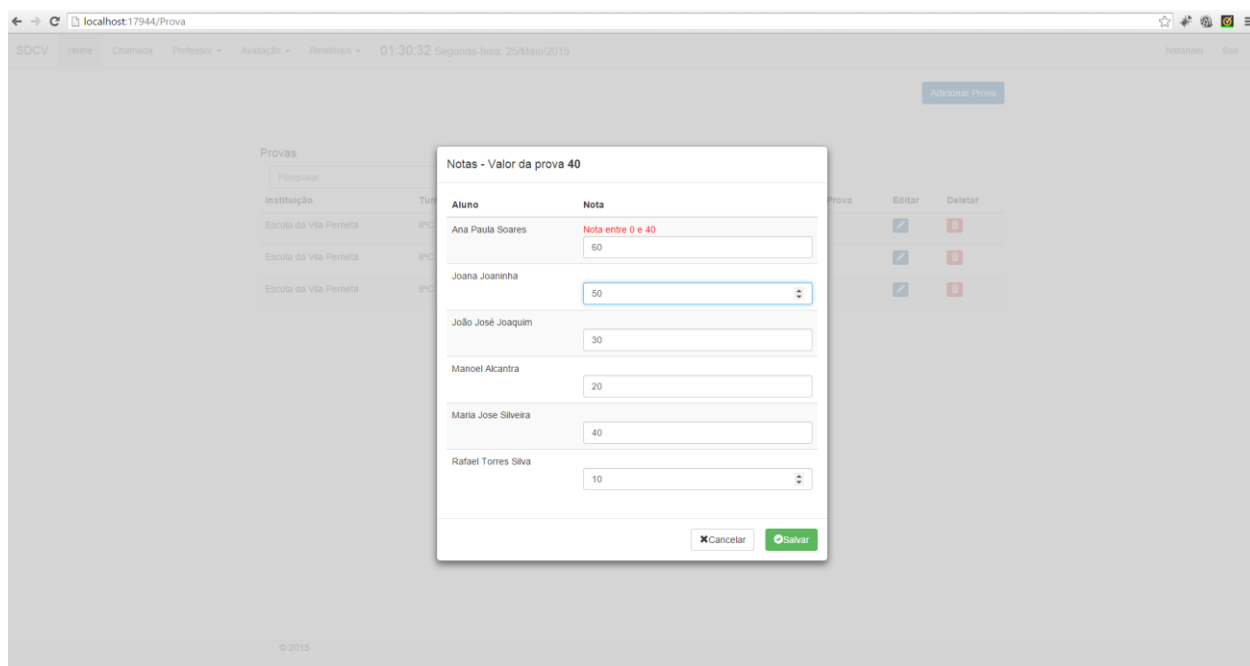


Figura 19. Tela modal de notas.

Ao ser aberto, esse modal de notas permite ao professor atribuir a nota da prova a cada aluno da turma. Após preencher o campo NOTA para todos os alunos, o docente clica em SALVAR, para que as informações possam ser inseridas no sistema. Também é possível abortar a ação através do botão CANCELAR de acordo com o modal da figura 19.

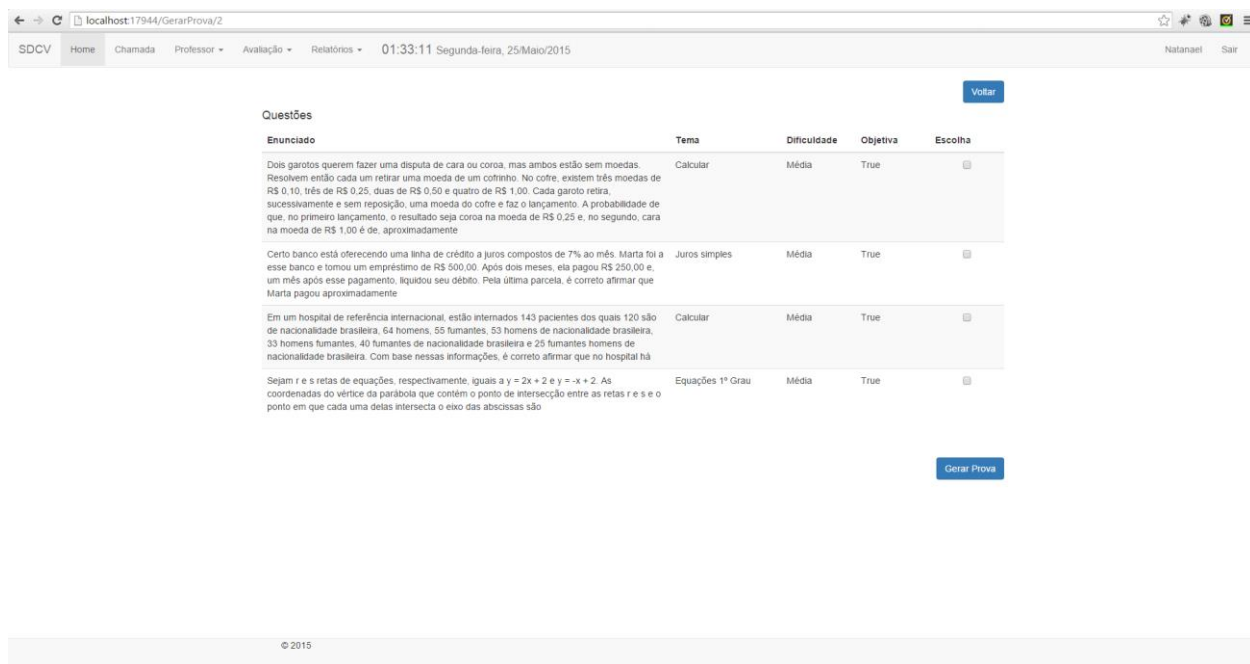


Figura 20. Tela de geração de prova.

Na tela da figura 20, são listadas todas as questões cadastradas, para que o professor possa escolher as que vão compor a prova gerada. Para escolher as questões que serão organizadas em formato de prova, o professor deve selecionar o checkbox da coluna ESCOLHA, referente a questão desejada. Após as escolhas terem sido concluídas, clica-se no botão gerar prova para que o sistema possa redirecionar para o arquivo montado (figura 21). É possível ainda voltar à pagina do CRUD através do botão VOLTAR.

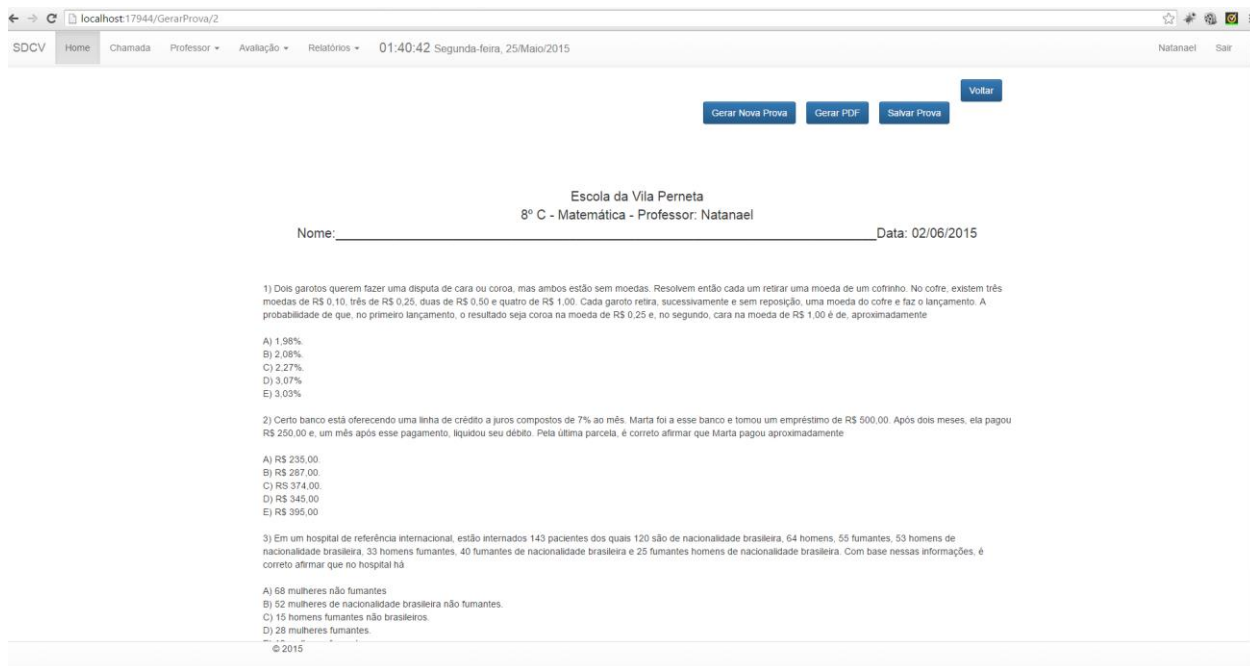


Figura 21. Prova gerada.

A figura 21 mostra a prova gerada conforme as questões escolhidas na tela anterior (figura 20). O docente possui as opções de exportar a prova para arquivo não editável (\*.pdf) clicando em gerar PDF, VOLTAR para a tela de escolha de questões ou SALVAR PROVA gerada, no sistema.

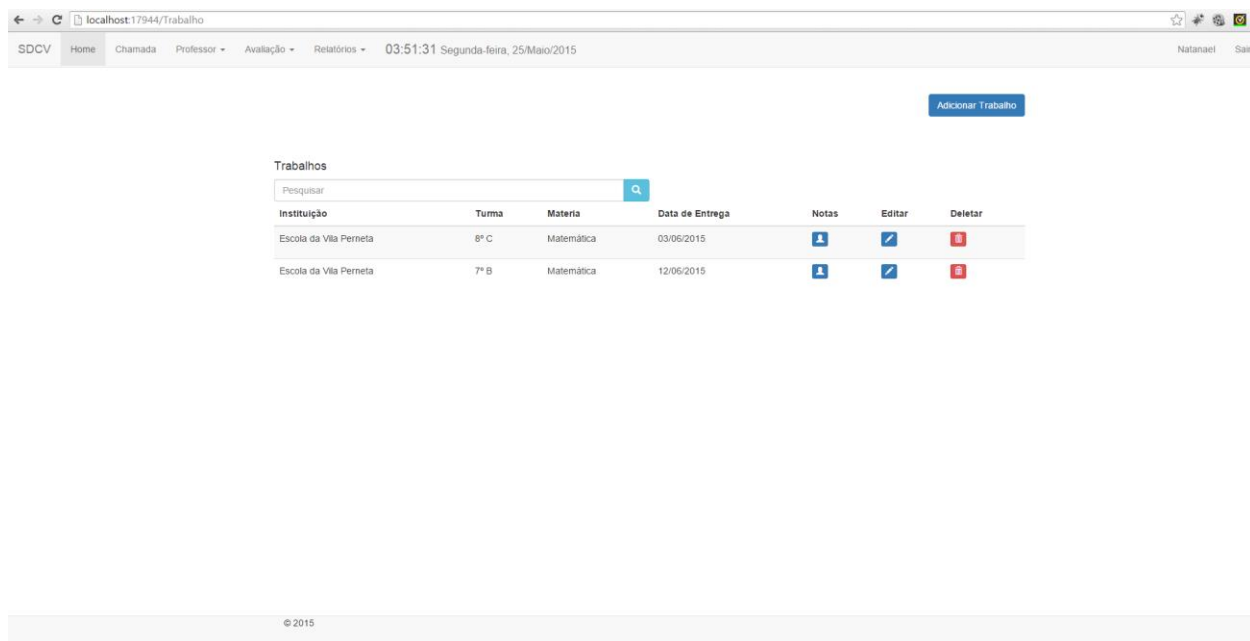


Figura 22. CRUD de Trabalho

No CRUD de trabalho, o educador pode pesquisar os trabalhos cadastrados no sistema de diário de classe virtual por partes do INSTITUIÇÃO, MATÉRIA ou por TURMA. As operações permitidas na tela são EDITAR ou DELETAR informações sobre o trabalho. Ao clicar em ADICIONAR TRABALHO, abre-se uma modal para a inserção de novo trabalho no sistema, similar aos anteriores descritos neste trabalho, em que o usuário deve preencher os campos e salvar os dados. se clicado, o ícone da coluna NOTAS do trabalho em questão abre um modal para atribuir aos alunos, as notas referentes ao trabalho em questão. (figura 22).

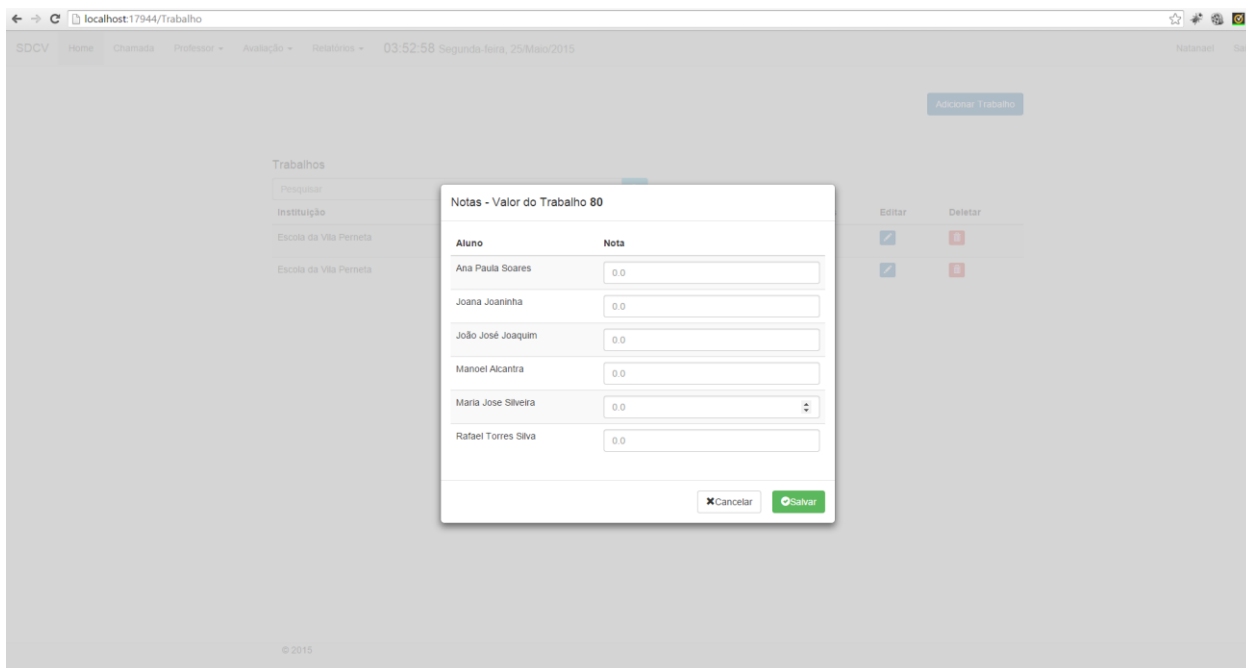


Figura 23. Tela modal de notas do trabalho.

Neste modal, o professor atribui a cada aluno da turma, a nota referente ao trabalho em questão, preenchendo o campo da coluna NOTA referente ao aluno desejado. Concluídas as atribuições, o professor pode clicar em SALVAR para enviar os dados para o sistema. Também é permitido o aborto da ação, através do botão CANCELAR aquiesça figura 23.



SDCV	Home	Chamada	Professor	Avaliação	Relatórios	03:54:04 Segunda-feira, 25/Maio/2015	Natanael	Sair
------	------	---------	-----------	-----------	------------	--------------------------------------	----------	------

Relatorio de Presença										
8º C - Matemática - Escola da Vila Permetá										Exportar em Excel
Nome	01/06/2015	03/06/2015	05/06/2015	10/06/2015	19/06/2015	27/06/2015	chamadas	Presença	faltas	Presença
Ana Paula Soares	✓	✓	✓	✓	✓	✓	6	6	X	100%
Joana Joaquina	✓	✓	✓	✓	X	✓	6	5	✓	83%
João José Joaquim	X	X	X	✓	✓	✓	6	3	3	50%
Manoel Alcantra	X	✓	X	X	✓	✓	6	3	3	50%
Maria Jose Silveira	✓	✓	✓	✓	✓	X	6	5	✓	83%
Rafael Torres Silva	✓	✓	✓	✓	X	✓	6	5	✓	83%

© 2015
--------

Figura 24. Tela de relatório de presença.

Esta tela (figura 23) é acessada pelo menu superior, e que gera o relatório de frequência da turma, indicando quantidade de presenças e faltas, bem como a porcentagem de presenças, indicando em cores (verde e vermelho) os alunos aprovados ou não por frequência, respectivamente. Ao clicar na opção EXPORTAR EM EXCEL, o sistema gera uma planilha eletrônica com o relatório em questão.

Relatório de Notas

8º C - Matemática - Escola da Vila Perna

Exportar em Excel

Alunos:	Valor: 40	Valor: 100	Valor: 60
Nome	02/06/2015	24/06/2015	25/06/2015
Ana Paula Soares	31	60	60
Joana Joaquina	20	80	50
João José Joaquim	30	50	23
Manoel Alcantra	20	60	58
Maria Jose Silveira	40	100	30
Rafael Torres Silva	10	12	10

© 2015

Figura 25. Tela de relatório de notas.

a tela identificada na figura 25 também pode ser acessada a partir do menu superior, e gera um relatório com todas as notas das provas dos alunos, de acordo com a TURMA selecionada. Apresenta a opção de EXPORTAR EM EXCEL, para enviar o relatório para um arquivo de planilha eletrônica.

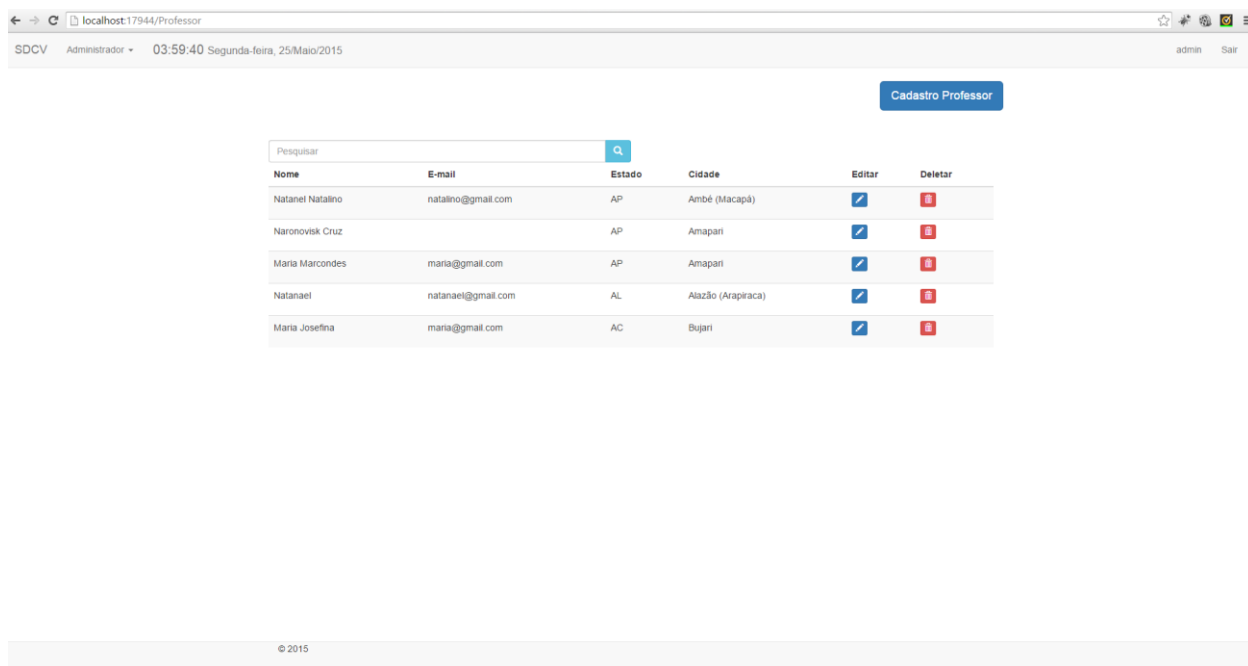


Figura 26. Tela Inicial do administrador.

Se o administrador do sistema acessa o sistema utilizando a tela de login única (figura 2) o sistema é redirecionado para a tela inicial do administrador, que tem função de um CRUD de professores, pesquisando, editando, excluindo (figura 26) e adicionando professores para utilizar o sistema (figura 27).

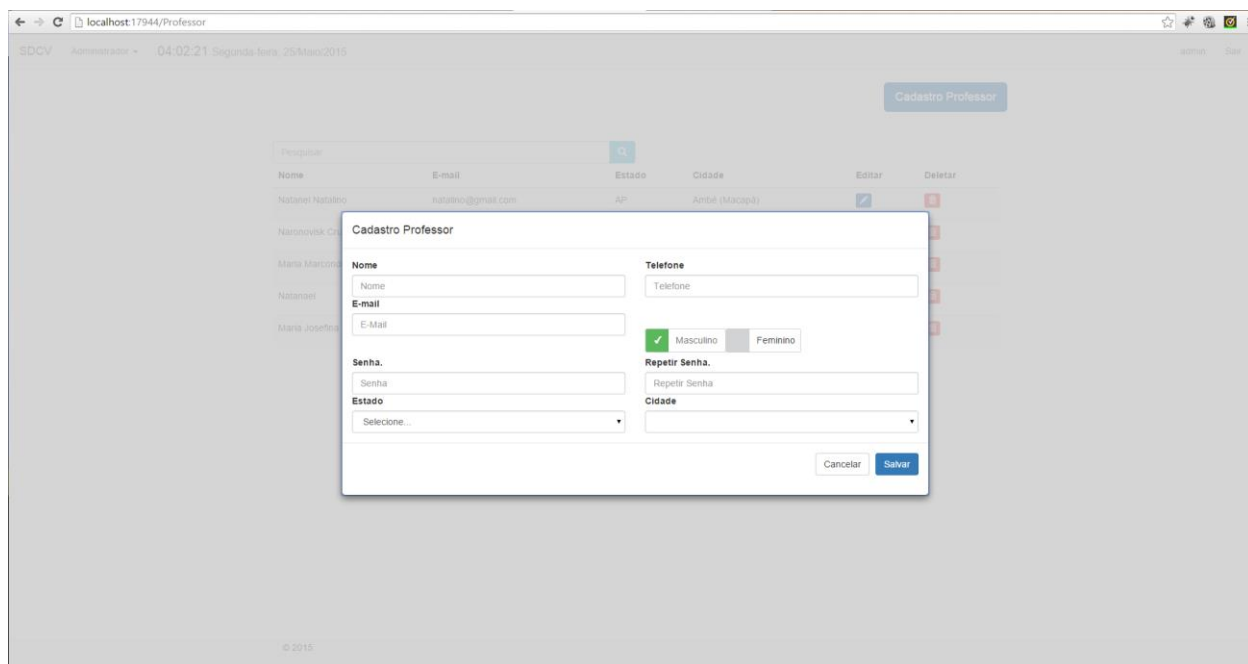


Figura 27. Tela modal de cadastro de professores para administrador.

Por meio do modal indicado pela figura 27, o administrador do sistema consegue cadastrar um professor, mediante o preenchimento dos campos NOME, EMAIL, TELEFONE, SEXO, SENHA, REPETIR SENHA, ESTADO e CIDADE. Após, salva-se os dados do novo professor clicando no botão SALVAR. Posteriormente ao registro, o novo professor cadastrado está apto a acessar o sistema de diário de classe virtual. É possível abortar a ação clicando em CANCELAR.

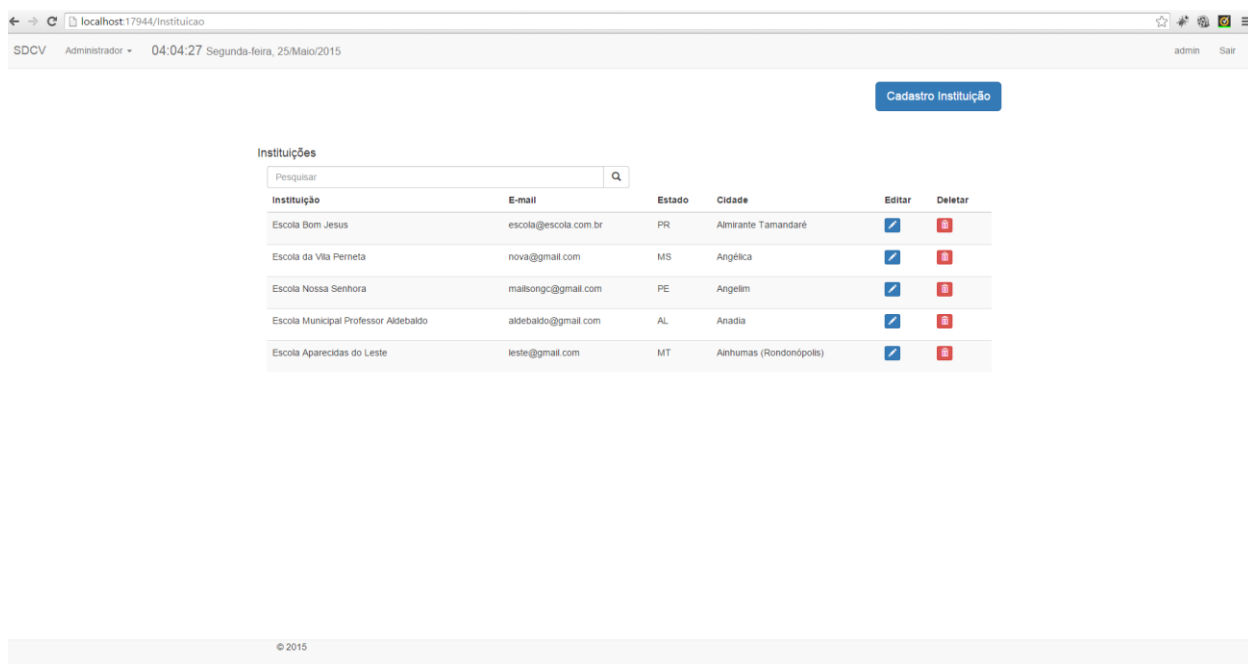


Figura 28. CRUD de Instituição.

Para que os professores possam usar as instituições de ensino no sistema, estas devem estar devidamente cadastradas. Neste CRUD de instituições, o administrador faz o gerenciamento das instituições, pesquisando, editando e excluindo instituições pesquisadas (figura 28) e ao clicar em ADICIONAR INSTITUIÇÕES, um modal é aberto para que se entre com as informações da nova instituição de ensino (figura 29).

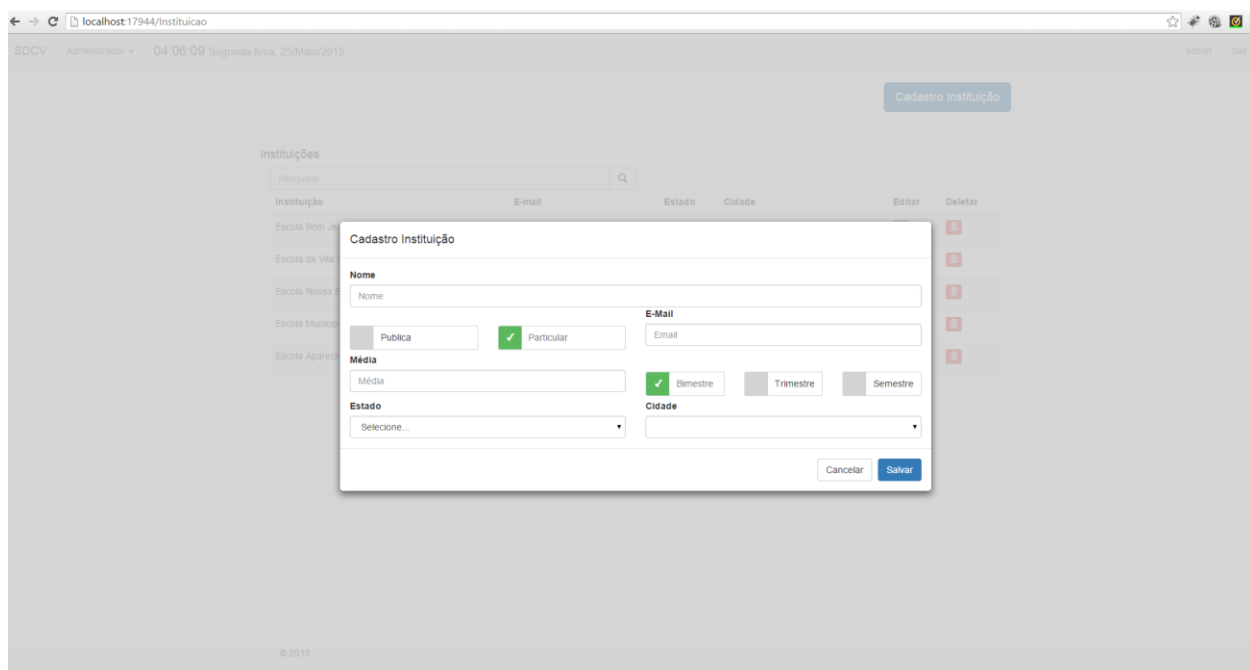


Figura 29. Tela modal de cadastro de instituição.

Na tela modal de cadastro de instituição representada pela figura 29, o administrador preenche os campos NOME, EMAIL, MEDIA, ESTADO, CIDADE, TIPO DE INSTITUIÇÃO e PERÍODO LETIVO para SALVAR e disponibilizar a instituição de ensino no sistema. É possível abortar a ação, clicando em CANCELAR.

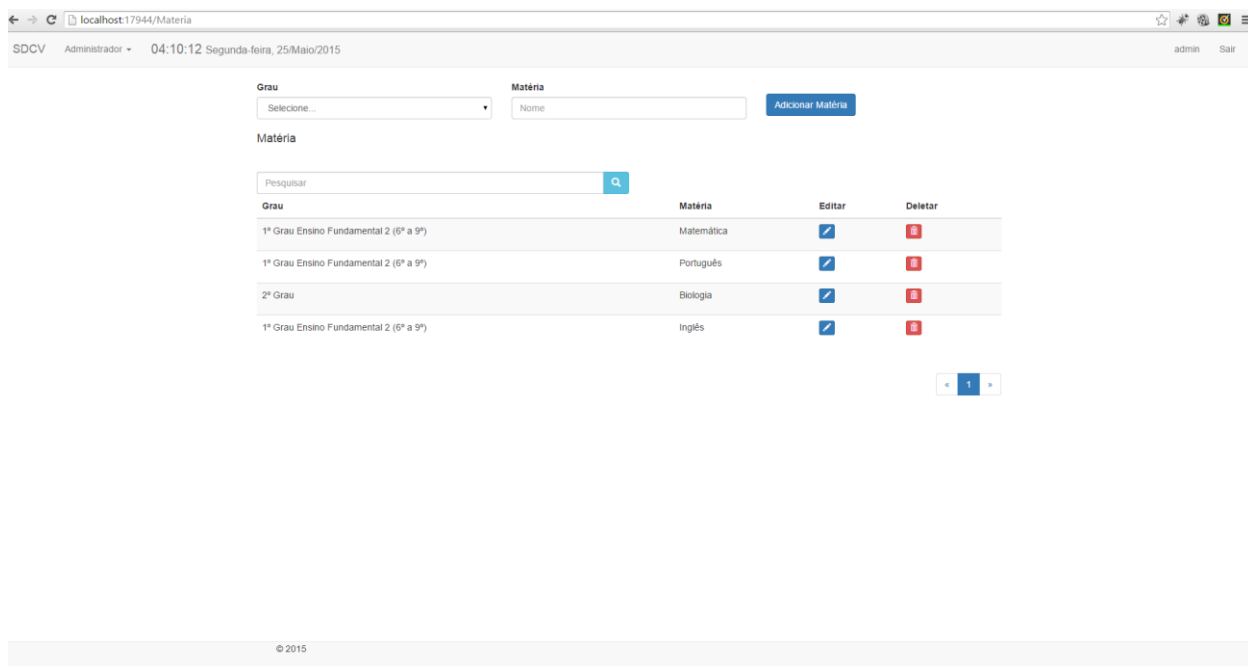


Figura 30. CRUD de Matéria

Esta tela (figura 30) também traz funcionalidades a nível de administrador, e permite pesquisar (por GRAU/MATÉRIA), editar e deletar as matérias (disciplinas). Permite ainda o cadastro de nova disciplina clicando no botão ADICIONAR MATÉRIA, que abre um modal de cadastro semelhante aos já mencionados, para salvar as informações da matéria, disponibilizando assim para uso no sistema.

SDCV Administrador 04:11:51 Segunda-feira, 25/Maio/2015 admin Sair

Relatorio de Instituções

Selecione...

Selecione

Escola Bom Jesus

Escola da Vila Perneta

Escola Nossa Senhora

Escola Municipal Professor Aldebaldo

Escola Aparecidas do Leste

Exportar em Excel

				Professor
				Nataniel Natalino
				Nataniel
Escola da Vila Perneta	Matemática	7º B	70	Nataniel
Escola da Vila Perneta	Matemática	8º D	70	Nataniel
Escola da Vila Perneta	Matemática	8º D	70	Nataniel
Escola da Vila Perneta	Matemática	8º D	70	Nataniel Natalino
Escola da Vila Perneta	Matemática	5º A	70	Nataniel Natalino
Escola Municipal Professor Aldebaldo	Matemática	5º A	60	Nataniel
Escola Municipal Professor Aldebaldo	Matemática	6º B	60	Nataniel
Escola Municipal Professor Aldebaldo	Matemática	5º B	60	Nataniel
Escola Municipal Professor Aldebaldo	Matemática	6º A	60	Nataniel

© 2015

Figura 31. Tela de relatório administrativo.

Acessado pelo menu superior, a tela de relatório administrativo da figura 31 fornece todos os dados das aulas organizado por instituição, contendo com nome do PROFESSOR, MATERIA e TURMA a qual da aula, além da MÉDIA adotada pela escola. O administrador pode clicar em EXPORTAR EM EXCEL se desejar gerar o arquivo de planilha eletrônica para o relatório.



## 5. TESTES

Segundo Neto, 2008, teste de software:

é o processo de execução de um produto para determinar se ele atingiu suas especificações e funcionou corretamente no ambiente para o qual foi projetado. O seu objetivo é revelar falhas em um produto, para que as causas dessas falhas sejam identificadas e possam ser corrigidas pela equipe de desenvolvimento antes da entrega final.

Para este mesmo autor, os testes podem ser classificados em 5 tipos:

- Teste de unidade: verifica se há erros no modulo;
- Teste de integração: visa encontrar bugs na comunicação entre os módulos;
- Teste de sistema: procura falhas no sistema (simulando o usuário);
- Teste de aceitação: realizado por um grupo de usuários finais; e
- Teste de regressão: repete os testes já aplicados no sistema a cada nova versão para evitar erros em módulos estáveis.

Existem duas formas diferentes de aplicar os testes acima descritos: os testes estruturais (chamados de caixa branca) e os funcionais (caixa preta). A diferença entre os dois consiste em conhecer a estrutura interna do software – no primeiro caso – e não conhecer a sua estrutura – o segundo caso – sendo que os testes caixa preta são focados em entradas e saídas, não levando em consideração o processamento da informação (Neto, 2008).

Nosso trabalho utilizou ambos os testes, sendo aplicado o teste caixa branca no caso de uso do login, e os dois testes no caso de uso referente a manter professor (mais especificamente a tela de cadastro). A seguir os resultados obtidos:

### **Teste caixa-branca.**

#### **Método do caminho básico.**

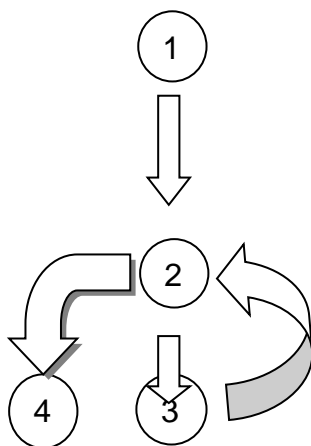
```
<div class="row">  
<div class="col-sm-12 col-md-10 col-md-offset-1 ">  
<div class="form-group">  
<div class="input-group">
```

```

<span class="input-group-addon">
<i class="glyphicon glyphicon-user"></i>
</span>
① <asp:TextBox ID="txtEmail" runat="server" CssClass="form-control"
placeholder="E-mail"></asp:TextBox>
</div>
</div>
<div class="form-group">
<div class="input-group">

<span class="input-group-addon">
<i class="glyphicon glyphicon-lock"></i>
</span>
② <asp:TextBox ID="txtSenha" runat="server" placeholder="Senha"
CssClass="form-control" TextMode="Password" />
</div>
</div>
<asp:Label ID="lblAviso" Font-Bold="true" ForeColor="Red"
runat="server"></asp:Label>
<div class="form-group">
③ <asp:LinkButton runat="server" ID="btnLogin" type="submit"
OnClick="btnLogin_Click1" CssClass="btn btn-lg btn-primary btn-block">
Entrar</asp:LinkButton>
</div>
④
</div>
</div>

```



$$V(g) = E - n + 2$$

$$V(g) = 4 - 4 + 2$$

$$V(g) = 2$$

Caminho 1,2,4.

### 1.Caso de teste

**Descrição:** Email e senha correspondentes

**Procedimento:** Inserir valores no campo email no formato de email [exemplo@ex.com](mailto:exemplo@ex.com), com a senha correta.

**Resultado Esperado:** Não deve aparecer nenhuma mensagem de advertência.

Caminho 1,2,3,4

## 2.Caso de teste

**Descrição:** Email e senha invalidos.

**Procedimento:** Inserir dados correspondentes e-mail e senha.

**Resultado Esperado:** Aparecer a mensagem "Email ou senha incorretos."

```
protected void btnLogin_Click1(object sender, EventArgs e)
{
    1 BLL.Professor oProfessor = BLL.Professor.Login(txtEmail.Text,
txtSenha.Text);
    6 if (oProfessor != null)
    {
        7 Session["Usuario"] = oProfessor.Nome;
        Session["idUsuario"] = oProfessor.IdfProfessor;
        Session["Admin"] = oProfessor.Administrador;

        8 if(!oProfessor.Administrador)
        {
            9 Response.Redirect(Page.GetRouteUrl("Home", null));
        }
        1 Response.Redirect(Page.GetRouteUrl("Professor", null));
    }
    else
    {
        1 lblAviso.Text = "Email ou senha incorretos.";
    }
}
```

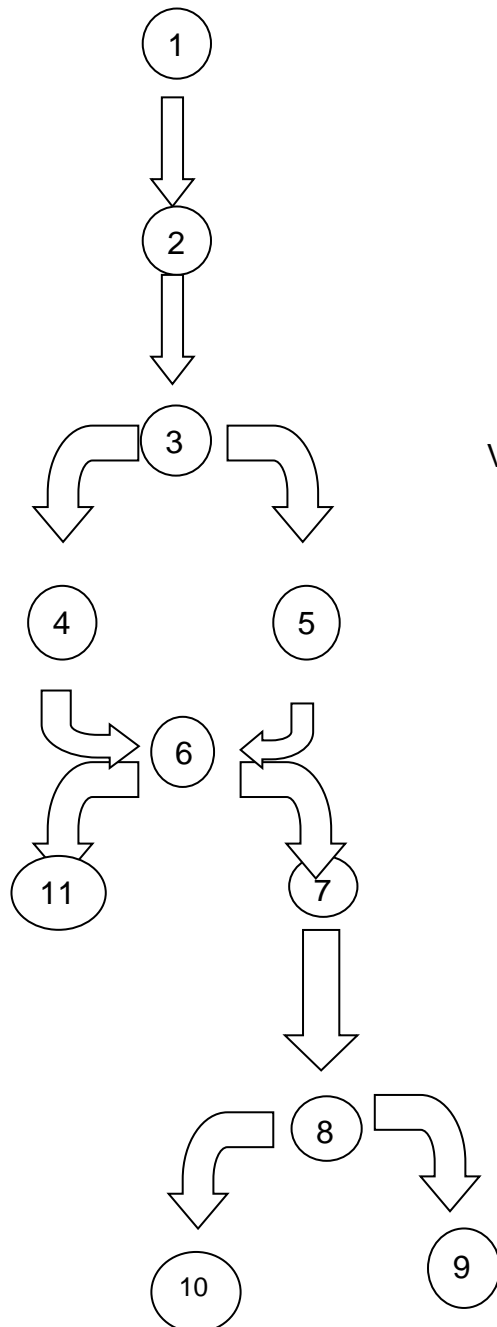
```
public static Professor Login(string email, string senha)
{
    List<SqlParameter> parm = new List<SqlParameter>();
    parm.Add(new SqlParameter("@email", SqlDbType.VarChar, 60));
    parm.Add(new SqlParameter("@senha", SqlDbType.VarChar, 20));
    parm[0].Value = email;
    parm[1].Value = senha;

    2 using (IDataReader dr =
DataAccessLayer.ExecuteReader(CommandType.Text, spLogin, parm))
    {
        Professor objProfessor = new Professor();
        3 if (SetInstance(dr, objProfessor))
        {
            4 return objProfessor;
        }
    }
}
```

```

    }
    5 return null;
  }

```



$$V(g) = E - n + 2$$

$$V(g) = 13 - 11 + 2$$

$$V(g) = 4$$

Caminho 1 – 1,2,3,5,6,11

# 1. Caso de teste.

**Descrição:** Não conseguiu acessar o bando de dados

**Procedimento:** Inserir usuário “[Mailson@ex.com](mailto:Mailson@ex.com)” e senha “123456”

**Resultado Esperado:** Error na pagina “Server Error in '/' Application”.

Caminho 2 – 1,2,3,5,6,11

## 2. Caso de teste.

**Descrição:** Email incorreto ou senha incorretos

**Procedimento:** Inserir email e senha não correspondentes ou não cadastrado ex. [aaaaa@oiejf.com](mailto:aaaaa@oiejf.com).

**Resultado Esperado:** Mensagem de “Email ou senha incorretos”

Caminho 3 – 1,2,3,4,6,7,8,9

## 3. Caso de teste

**Descrição:** Login de professor.

**Procedimento:** Na entrada de email e senha utilizar as mesmas correspondes e possuir cadastro.

**Resultado Esperado:** Efetuar o login

Caminho 4 – 1,2,3,4,6,8,10

## 4. Caso de teste

**Descrição:** Login efetuado com sucesso

**Procedimento:** Na entrada de email e senha utilizar as mesmas correspondes e possuir cadastro.

**Resultado Esperado:** Efetuar o login.

## Método do caminha básico.

```
<asp:Panel ID="pnlCadastro" runat="server" Style="display: none">
  <!-- Modal -->
  <div class="bs-example-modal-lg" tabindex="2" role="dialog" aria-
labelledby="myModalLabel" aria-hidden="true">
    <div class="modal-dialog modal-lg">
      <div class="modal-content">
        <div class="modal-header">
          <%-- <button type="button" class="close" data-dismiss="modal"
aria-label="Close"><span aria-hidden="true">&times;</span></button>--%>
          <h4 class="modal-title" id="myModalLabel">Cadastro
Professor</h4>
        </div>
        <div class="modal-body">
          <form class="form-horizontal">
            <asp:UpdatePanel ID="upCadastro" runat="server"
UpdateMode="Conditional">
              <ContentTemplate>
                <%--Formulario de cadastro usuario--%>
                <div class="row">
```

```

        <div class="col-md-6">
            <div class="control-group">
                <label class="control-label"
for="textinput">Nome</label>
                2 <asp:RequiredFieldValidator runat="server"
ControlToValidate="txtNome" ForeColor="red" ValidationGroup="Cadastro"
ErrorMessage="Campo Obrigatorio"></asp:RequiredFieldValidator>
                <div class="controls">
                    1 <asp:TextBox ID="txtNome" MaxLength="70"
placeholder="Nome" CssClass="form-control" runat="server" data-error="Campo
Obrigatorio"></asp:TextBox>
                </div>
            </div>
        </div>

        <div class="col-md-6">
            <div class="control-group">
                <label class="control-label"
for="textinput">Telefone</label>
                3 <asp:RequiredFieldValidator runat="server"
ControlToValidate="txtTelefone" ForeColor="Red" ValidationGroup="Cadastro"
ErrorMessage="Campo Obrigatorio"></asp:RequiredFieldValidator>
                <div class="controls">
                    4 <asp:TextBox ID="txtTelefone" MaxLength="70"
placeholder="Telefone" CssClass="form-control" runat="server" data-error="Campo
Obrigatorio"></asp:TextBox>
                </div>
            </div>
        </div>
    </div>
    <div class="row">
        <div class="col-md-6">
            <div class="control-group">
                5 <label class="control-label" for="textinput">E-
mail</label><asp:RegularExpressionValidator ID="validEmail" runat="server"
ControlToValidate="txtEmailCadastro"
                    ErrorMessage=" E-mail Invalido"
SetFocusOnError="True" ValidationExpression="\w+([-+.']\w+)*@\w+([-+.\w+([-
.]\w+)*"
                    ForeColor="Red"
ValidationGroup="Cadastro"></asp:RegularExpressionValidator>
                <asp:RequiredFieldValidator runat="server"
ControlToValidate="txtEmailCadastro" ForeColor="Red" ValidationGroup="Cadastro"
ErrorMessage="Campo Obrigatorio"></asp:RequiredFieldValidator>
                <div class="controls">
                    6

```

```

        <asp:TextBox ID="txtEmailCadastro" type="email"
MaxLength="70" placeholder="E-Mail" CssClass="form-control"
runat="server"></asp:TextBox>
    </div>
</div>
</div>
<div class="col-md-6">
    <div class="control-group">
        7 <asp:RequiredFieldValidator runat="server"
ControlToValidate="rbSexo" ForeColor="Red" InitialValue=""
ValidationGroup="Cadastro" ErrorMessage="Campo
Obrigatorio"></asp:RequiredFieldValidator>
        8 <asp:RadioButtonList runat="server"
RepeatDirection="Horizontal" class="funkyradio-success funkyradio" ID="rbSexo">
            <asp:ListItem Value="M">Masculino
        &nbsp;&nbsp;&nbsp;</asp:ListItem>
            <asp:ListItem Value="F">Feminino
        &nbsp;</asp:ListItem>
        </asp:RadioButtonList>
    </div>
</div>
</div>
<div class="row">
    <div class="col-md-6">
        <!-- Text input-->
        <div class="control-group">
            <label class="control-label"
for="textinput">Senha.</label>
            9 <asp:RequiredFieldValidator runat="server"
ControlToValidate="txtSenhaCadastro" ForeColor="Red"
ValidationGroup="Cadastro" ErrorMessage="Campo
Obrigatorio"></asp:RequiredFieldValidator>
            <div class="controls">
                10 <asp:TextBox ID="txtSenhaCadastro"
MaxLength="20" TextMode="Password" runat="server" placeholder="Senha"
CssClass="form-control"></asp:TextBox>
            </div>
        </div>
    </div>
    <div class="col-md-6">
        <!-- Text input-->
        <div class="control-group">
            <label class="control-label" for="textinput">Repetir
Senha.</label>
            <asp:RequiredFieldValidator runat="server"
ControlToValidate="txtReSenha" ForeColor="Red" ValidationGroup="Cadastro"
ErrorMessage="Campo Obrigatorio"></asp:RequiredFieldValidator>

```

```

11      <asp:CompareValidator ID="comparePasswords"
        runat="server" ForeColor="Red"
        ControlToCompare="txtSenhaCadastro"
        ControlToValidate="txtReSenha"
        ErrorMessage="Senhas são diferentes."
        Display="Dynamic" />
12      <div class="controls">
        <asp:TextBox ID="txtReSenha" MaxLength="20"
TextMode="Password" runat="server" placeholder="Repetir Senha"
CssClass="form-control"></asp:TextBox>
      </div>
    </div>
  </div>
  <asp:UpdatePanel ID="upCidade"
UpdateMode="Conditional" runat="server">
    <ContentTemplate>
      <div class="row">
        <div class="col-md-6">
          <div class="control-group">
            <label class="control-label"
for="selectbasic">Estado</label>
1          <asp:RequiredFieldValidator runat="server"
ForeColor="Red" ControlToValidate="ddlEstado" InitialValue="0"
ValidationGroup="Cadastro" ErrorMessage="Campo
Obrigatorio"></asp:RequiredFieldValidator>
            <div class="controls">
              <asp:DropDownList ID="ddlEstado"
AutoPostBack="true" runat="server" name="selectbasic"
OnSelectedIndexChanged="ddlEstado_SelectedIndexChanged" class="form-
control">
14              </asp:DropDownList>
            </div>
          </div>
        </div>
        <div class="col-md-6">
          <!-- Select Basic -->
          <div class="control-group">
            <label class="control-label"
for="selectbasic">Cidade</label>
15          <asp:RequiredFieldValidator runat="server"
ControlToValidate="ddlCidade" ForeColor="red" InitialValue="0"
ValidationGroup="Cadastro" ErrorMessage="Campo
Obrigatorio"></asp:RequiredFieldValidator>
            <div class="controls">

```



```

                                <asp:DropDownList runat="server"
ID="ddlCidade" name="selectbasic" class="form-control">
                                1 </asp:DropDownList>
                                </div>
                                </div>
                                </div>
                                </div>
                                </ContentTemplate>
                                </asp:UpdatePanel>

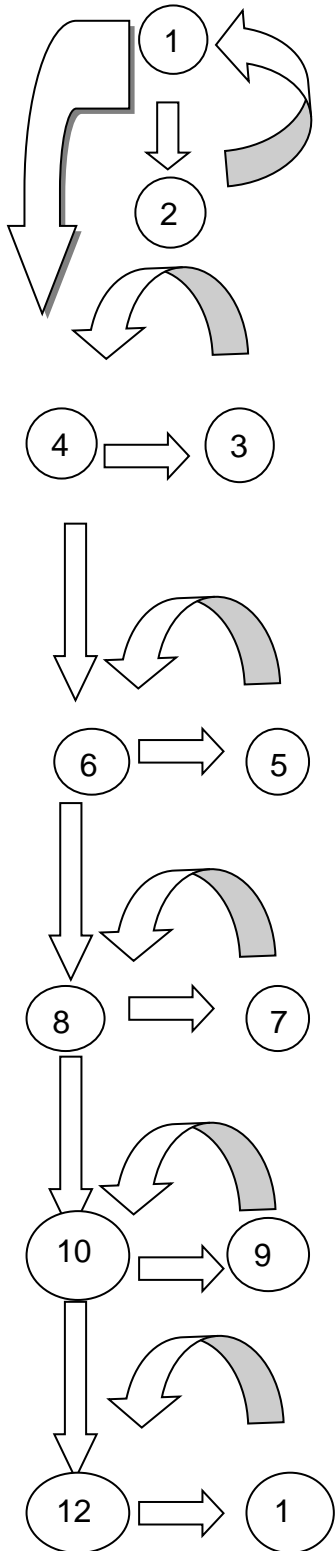
                                <%--Fim cadastro Usuario--%>
                                </ContentTemplate>
                                </asp:UpdatePanel>

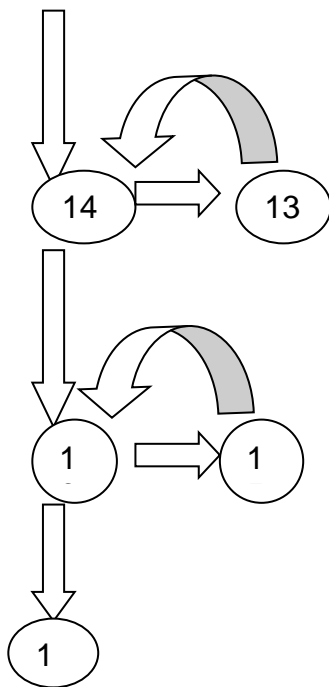
                                <!-- Button -->

                                </form>
                                </div>
                                <div class="modal-footer">
                                1 <div class="form-group">
                                <asp:Button runat="server" ID="btnClose" class="btn btn-default"
Text="Cancelar" data-dismiss="modal"></asp:Button>
                                <asp:Button runat="server" ID="btnCadastrar" class="btn btn-
primary" 1 "Salvar" ValidationGroup="Cadastro" OnClick="btnCadastrar_Click" />
                                </div>
                                </div>
                                </div>
                                </div>
                                </div>
                                </asp:Panel>

```

17





Caminho 1,2.

#### 1. Caso de teste

**Descrição:** Nome é inválido

**Procedimento:** Deixar o campo em branco.

**Resultado Esperado:** Aparecer a mensagem “Campo Obrigatorio”.

Caminho 1,4,3.

#### 2. Caso de teste

**Descrição:** Telefone é inválido

**Procedimento:** Inserir valores no telefone de 0 a 9, no máximo 9 dígitos.

**Resultado Esperado:** Aparecer a mensagem “Campo Obrigatorio”.

Caminho 1,4,6,5.

#### 3. Caso de teste

**Descrição:** E-mail é inválido.

**Procedimento:** Inserir valores no campo email no formato de email “exemplo@ex.com”.

**Resultado Esperado:** Aparecer a mensagem “E-mail inválido”.

Caminho 1,4,6,8,7.

#### 4. Caso de teste

**Descrição:** Sexo é inválido.

**Procedimento:** Escolher entre um dos radio-button.

**Resultado Esperado:** Aparecer a mensagem “Campo Obrigatorio”.

Caminho 1,4,6,8,10,9.

#### **5. Caso de teste**

**Descrição:** Senha é invalido.

**Procedimento:** Deixar o campo em branco.

**Resultado Esperado:** Aparecer a mensagem “Campo Obrigatorio”.

Caminho 1,4,6,8,10,12,11.

#### **6. Caso de teste**

**Descrição:** Repetir senha é invalido.

**Procedimento:** A repetição da senha for diferente da digitada no campo senha.

**Resultado Esperado:** Aparecer a mensagem “Senhas são diferentes”.

Caminho 1,4,6,8,10,12,14,13.

#### **7. Caso de teste**

**Descrição:** Estado é invalido.

**Procedimento:** Selecionar um estado do dropdownlist

**Resultado Esperado:** Aparecer a mensagem “Campo Obrigatorio”.

Caminho 1,4,6,8,10,12,14,16,15.

#### **8. Caso de teste**

**Descrição:** Cidade é invalido.

**Procedimento:** Selecionar um estado do dropdownlist

**Resultado Esperado:** Aparecer a mensagem “Campo Obrigatorio”.

Caminho 1,4,6,8,10,12,14,16,18.

#### **9. Caso de teste**

**Descrição:** Campos validos.

**Procedimento:** Inserir dados de maneira correta

**Resultado Esperado:** Não deve aparecer nenhuma mensagem de advertência.

Caminho 17.

#### **10. Caso de teste**

**Descrição:** Fechar modal de cadastro.

**Procedimento:** clicar no botão cancelar

**Resultado Esperado:** fechar modal do cadastro.

```

protected void btnCadastrar_Click(object sender, EventArgs e)
{
    BLL.Professor objProfessor = new BLL.Professor();
    Cidade ocidade = new Cidade();
    objProfessor.Nome = txtNome.Text;
    objProfessor.Email = txtEmailCadastro.Text;
    objProfessor.Senha = txtReSenha.Text;
    objProfessor.Telefone = txtTelefone.Text;
    objProfessor.Sexo = Convert.ToChar(rbSexo.SelectedValue);
    objProfessor.IdfCidade = Convert.ToInt32(ddlCidade.SelectedValue);
    1 objProfessor.Insert(objProfessor);
    mpeCadastro.Hide();
    LimpaCampos();
    8 mpeDel.Show();
}

```

```

public bool Insert(Professor objProfessor)
{
    try
    {
        List<SqlParameter> parms = GetParameters();
        SetParameters(parms, objProfessor);

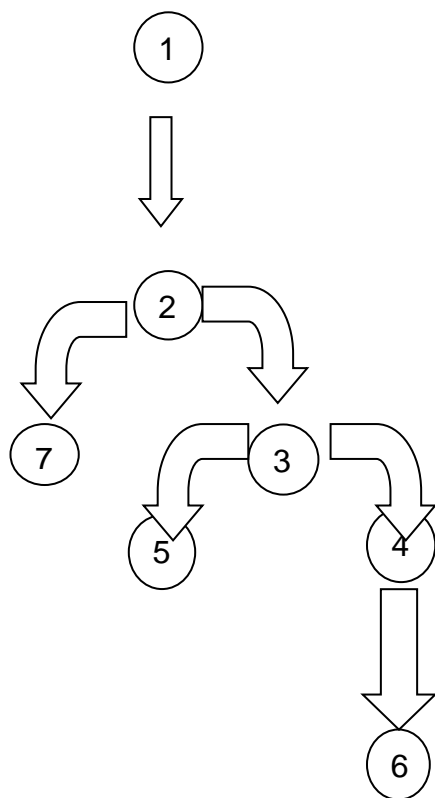
        using (SqlConnection conn = new
SqlConnection(DataAccessLayer.CONN_STRING))
        {
            2 conn.Open();
            using (SqlTransaction trans = conn.BeginTransaction())
            {
                try
                {
                    3 SqlCommand cmd =
DataAccessLayer.ExecuteNonQueryCmd(trans, CommandType.Text,
SPINSERT, parms);
                    cmd.Parameters.Clear();
                    trans.Commit();
                    objProfessor = Professor.Login(objProfessor.Email,
objProfessor.Senha);
                    4 Aula.CriarGrid(objProfessor.IdfProfessor);
                }
                catch

```

```

    {
      5 trans.Rollback();
        throw;
    }
  }
  6 return true;
}
catch
{
  7 return false;
}

```



Caminho 1 – 1,2,7

**1. Caso de teste.**

**Descrição:** Não conseguiu acessar o bando de dados

**Procedimento:** Abrir conexão com o banco

**Resultado Esperado:** Error na pagina “Server Error in '/' Application”.

Caminho 2 – 1,2,3,5

**2. Caso de teste.**

**Descrição:** Erro ao inserir no banco de dados

**Procedimento:** dado em formato incorreto.

**Resultado Esperado:** Mensagem de “Error”

Caminho 3 – 1,2,3,4,6

**3. Caso de teste**

**Descrição:** Inserir no banco de dados com sucesso.

**Procedimento:** dados de entrada corretos.

**Resultado Esperado:** Exibir a modal com a mensagem”Cadastro realiado com Sucesso”.

**4. Teste Caixa Preta.**

**Particionamento de equivalência.**

**Requisito:** o e-mail e senhas devem ser o mesmo dos cadastrados.

Classes de equivalência:

Os dois campo preenchidos de e-mail e senha.

Invalidas

Senha não correspondente com o e-mail

E-mail não correspondente com a senha.

**Requisito:** O telefone deve possuir 8 ou 9 dígitos e apenas números.

Classe de equivalência:

Quantidade de dígitos: menor que 8 ou maior que 9.

Nenhum dos dígitos ser letra.  
Invalidas..  
Quantidade de dígitos menor que 8.  
Quantidade de dígitos maior que 9.  
Digito possuir um ou mais letras.

**Requisito:** O DDD deve possuir 2 digitos apenas.

Classe de equivalência:

Quantidade de dígitos: igual a 2.  
Nenhum dos dígitos ser letra.

Invalidas.

Quantidade de dígitos menor que 2.  
Quantidade de dígitos maior que 2.  
Digito possuir um ou mais letras.

**Requisitos :** O email deve possuir “@” e “.” em sua composição.

Classe de equivalência.

Possuir @ em sua composição  
Possuir “.com”.

Invalidas.

Não possuir o digito “@”.  
Não possuir “.com “.

**Requisitos:** Login deve ser feito com email e senha cadastrados.

Classe de equivalência.

Possuir email cadastrado.  
Possuir senha cadastrada.

Invalidas.

Não possuir email cadastrado.  
Não possuir senha cadastrada.

**Requisitos:** Senha deve possuir no mínimo 4 dígitos e no Máximo 8 letras ou números.

Classe de equivalência.

Quantidade de dígitos menor que 8  
Quantidade de dígitos maior que 4.

Invalidas.

Quantidade de dígitos menor que 4  
Quantidade de dígitos maior que 8.

**Requisitos:** Campo Sexo ser escolhido dentre as duas opções.

Classe de equivalência.

Escolher uma opção.

Invalidas.

Deixar sem em branco.



**Requisitos:** Estado ser escolhido dentre os estados listados no dropdownlist.

Classe de equivalência.

Escolher uma opção.

Invalidas.

Deixar sem escolher (Selecione..).

**Requisitos:** Cidade ser escolhido dentre os estados listados no dropdownlist.

Classe de equivalência.

Escolher uma opção.

Invalidas.

Deixar sem escolher (Selecione..).

Ao aplicar os testes acima, foram detectados alguns bugs no sistema, que foram corrigidos e o sistema obteve êxito em uma reaplicação dos testes.

## 6. CONSIDERAÇÕES FINAIS

A proposta de LMS aqui apresentada sugere uma facilitação no gerenciamento das notas, frequência dos alunos e das atividades de sala de aula, o que possivelmente muito otimiza o tempo hábil do professor, isso acarretaria um possível aumento de desempenho do profissional da educação em sala de aula, além de incentivar o uso da tecnologia digital na educação.

O sistema deve ser adaptável às diferentes realidades no ensino, para que possa cumprir com o papel de “meio de incentivo” para a introdução de tecnologia em uma das áreas mais tradicionalistas de nossa sociedade. Entretanto, o *software* não está acessível para todos, pois há escolas que não possuem computadores disponíveis para os professores em sala de aula. Em outros lugares, a realidade nos mostra que há condições plenas para a implantação da LMS aqui apresentada, tornando viável o nosso esforço.

O desenvolvimento de software não é uma tarefa simples e exige o empenho e o preparo de todos os envolvidos. Nesse trabalho, houve satisfatório êxito nos processos empregados para desenvolver o software de diário de classe virtual, e, embora houve muita dificuldade no desenvolvimento devido à falta de tempo dos integrantes da equipe, ou mesmo a mudança de algumas tecnologias no decorrer do projeto.

## **AGRADECIMENTOS**

Ao Setor de Educação Profissional e Tecnológica da Universidade Federal do Paraná (SEPT/UFPR), por ceder o espaço físico onde o projeto foi desenvolvido. À cujo sem o seu apoio e auxílio, o trabalho não se realizaria. À professora Rafaela Mantovani Fontana, por todos os esclarecimentos teóricos prestados. Ao professor Luiz Antônio Acra pela concepção da ideia inicial.

## **REFERÊNCIAS**

- Bulegon, A. M.; Mussoi, E. M.; Tarouco, L. M. R.(2010). eXe Learning: Uma Ferramenta de Autoria para o Ensino e Aprendizagem. In: Salão de Ensino, 2010, Porto Alegre. Salão de Ensino 2010. UFRGS.
- Cantalice, L. M.(2002). Tecnologia na educação. Psicologia Escolar e Educacional, v. 6, n. 2, p. 187-187.
- Collison, S.(2006). Beginning CSS web development: from novice to professional. Apress.
- Damásio, M. J.(2007). Tecnologia e educação. As Tecnologias da Informação e da.
- De Moura Júnior, R. M. F. Conceitos Básicos de IHC. Recuperado em 02 de Maio 2015 de

<http://xa.yimg.com/kq/groups/46090643/1525393618/name/IFBaiano+-+IHC+-+Aula+01B+-+Conceitos+B%C3%A1sicos+IHC.pdf>.

D' Ignázio, F.(1992). Multimedia Sandbox. The Computing Teacher, 20 (2), 54-55.

Flanagan, D.(2004). JavaScript: O Guia Definitivo. 4ª ed, Porto Alegre, Bookman.

Freire, P.(1987). Pedagogia do Oprimido. Rio de Janeiro: Paz e terra.

Guedes, G. TA. (2009). UML 2. Uma Abordagem Prática, São Paulo, Novatec.

LABIUTIL-ERGOLIST/UFSC – Laboratório de Utilizabilidade. Universidade Federal de Santa Catarina. Recuperado em 01 de Abril de 2015 de <http://www.labiutil.inf.ufsc.br/ergolist/>.

Lévy, P. (1999). Cibercultura. Editora 34.

Lobato, L. L. et al. (2008). Novos estilos de interação em Sistemas de Gestão de Aprendizagem. In: Anais do XXVIII Congresso da SBC. p. 186-195.

Menezes, L. C. M.; Xavier, L. F. S.; Pereira, M. L. S. (2010). Gerenciamento do Escopo em Projetos. Ed. da FGV.

Merli, A. Stangherlim, R. Porque não registrar? A importância do registro como instrumento de reflexão na formação de professores. In: Anais IX Colóquio de Pesquisa sobre instituições escolares UNINOVE. Recuperado de [http://www.uninove.br/marketing?ix\\_coloquio/PDF/angelica\\_almeida.pdf](http://www.uninove.br/marketing?ix_coloquio/PDF/angelica_almeida.pdf) em 05 de Agosto de 2014.

Microsoft. Introdução à linguagem C# e ao .NET Framework, 2014. Recuperado de <https://msdn.microsoft.com/pt-br/library/z1zx9t92.aspx> em 22 de março de 2015.

Mukhi, V., Shanbhah, S., Mukhi S.(2002) C# - Fundamentos. São Paulo, Makron Books.

Neto, A. C. D.; Dias, C. Introdução a teste de software. Engenharia de Software Magazine, v. 1, 2008.

Raggett, Dave et al. (1999). HTML 4.01 Specification. W3C recommendation, v. 24.

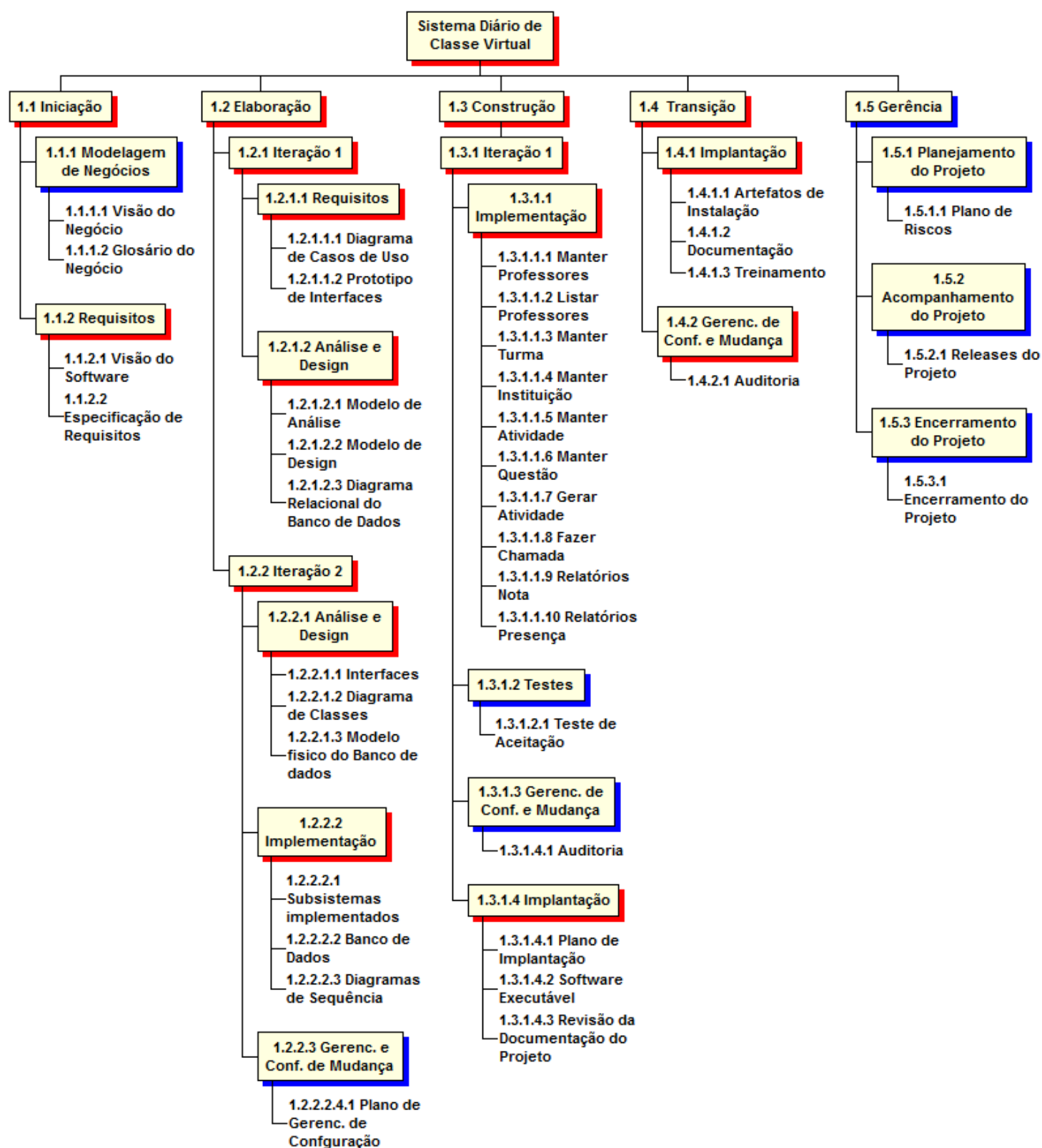
Shinkawa, Bárbara Poli Uliano, and Claudio Massaaki Shinkawa. "O docente e a organização temporal." *Revista Espaço Acadêmico* 14.162 (2014): 33-39.

Tarouco, L. M. R.; Fabre, M. C. J. M.; Grando, A. R. S.; Konrath, M. L. P. (2004). *Objetos de Aprendizagem para M-Learning*. Florianópolis: SUCESU - Congresso Nacional de Tecnologia da Informação e Comunicação. Recuperado em 15 de Setembro de 2014 de [http://www.cinted.ufrgs.br/CESTA/objetosdeaprendizagem\\_sucesu.pdf](http://www.cinted.ufrgs.br/CESTA/objetosdeaprendizagem_sucesu.pdf).

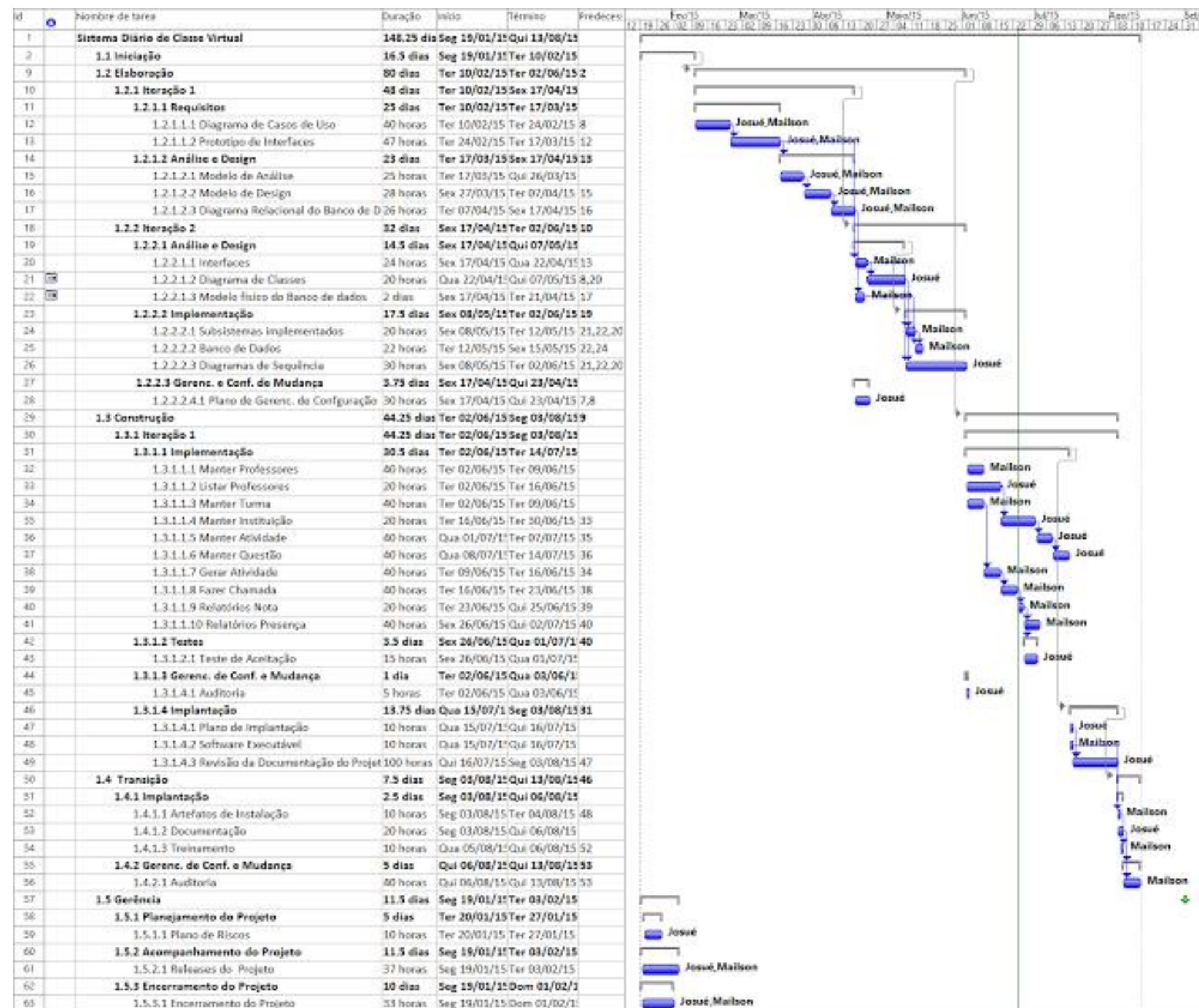
Tragtenberg, Maurício. "Relações de poder na escola." *Lua Nova: Revista de Cultura e Política* 1.4 (1985): 68-72.

Valente, J. A.; Almeida, F. J. de. Visão analítica da informática na educação no Brasil: a questão da formação do professor. *Revista Brasileira de Informática na Educação*, v. 1, n. 1, p. 45-60, 1997.

## APÊNDICE A – ESTRUTURA ANALÍTICA DO PROJETO (EAP)

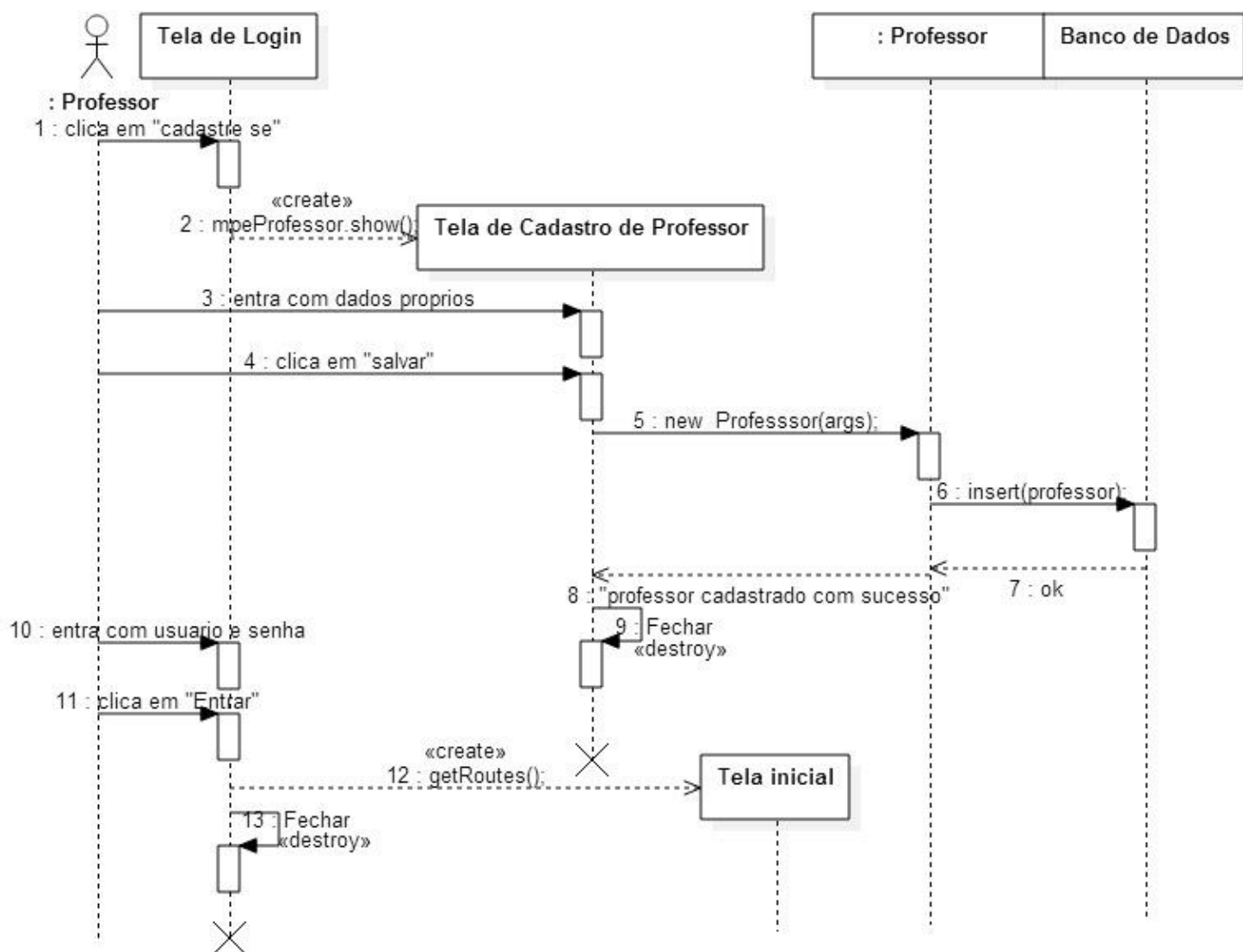


## APÊNDICE B – GRAFICO DE GANTT

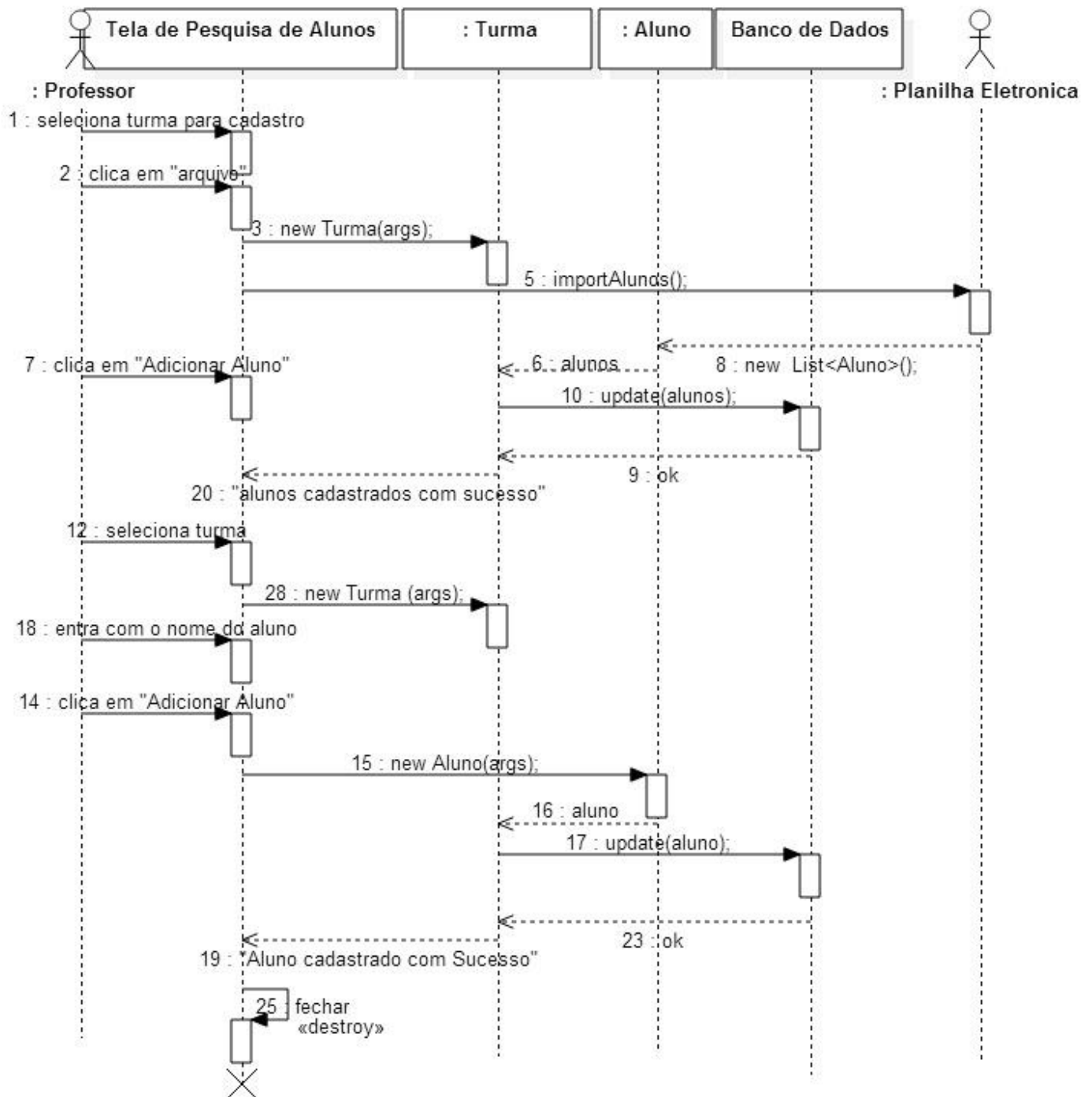


## APÊNDICE C – DIAGRAMAS DE SEQUENCIA

Cadastro de professor e login:

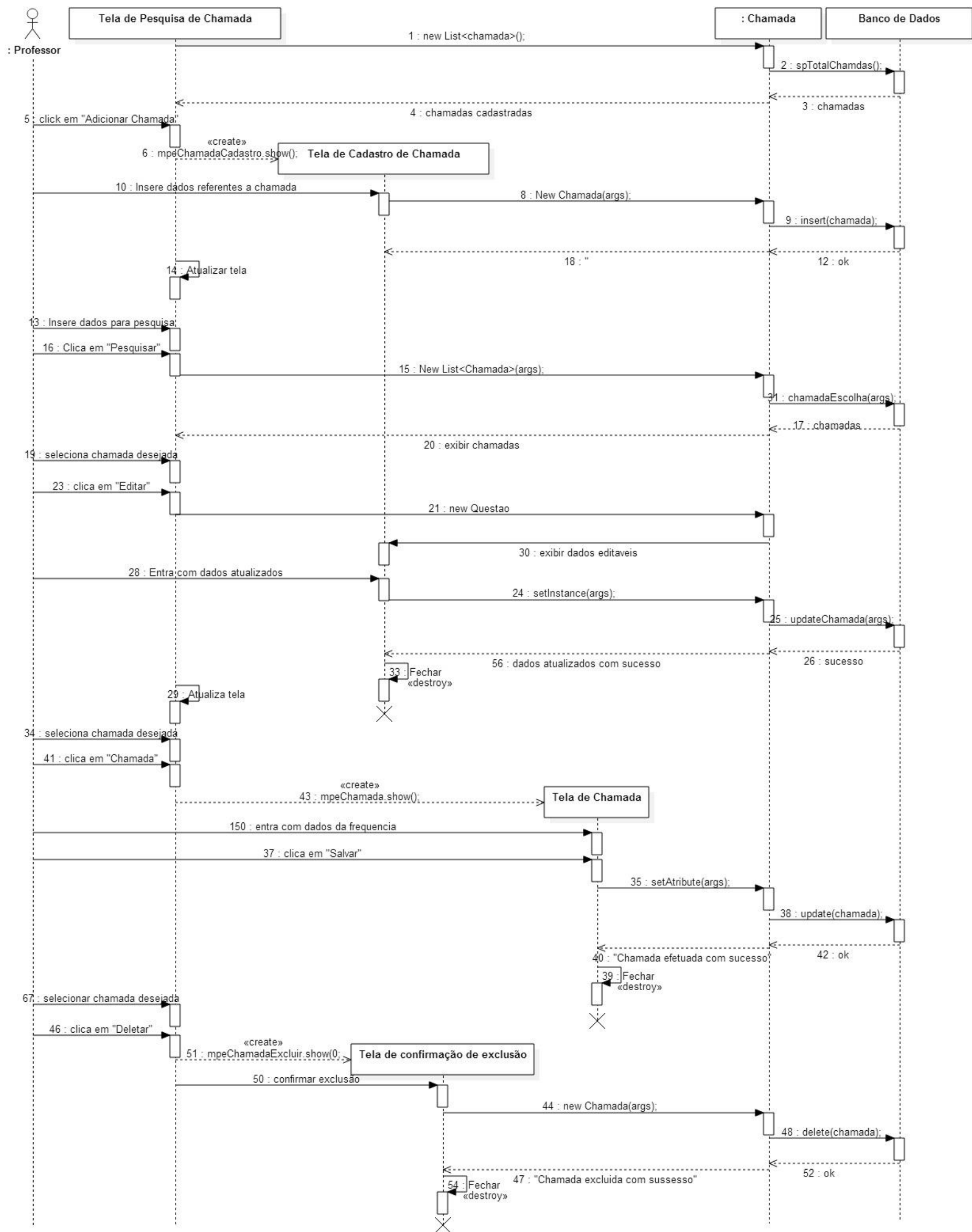


## Cadastro de alunos na turma:

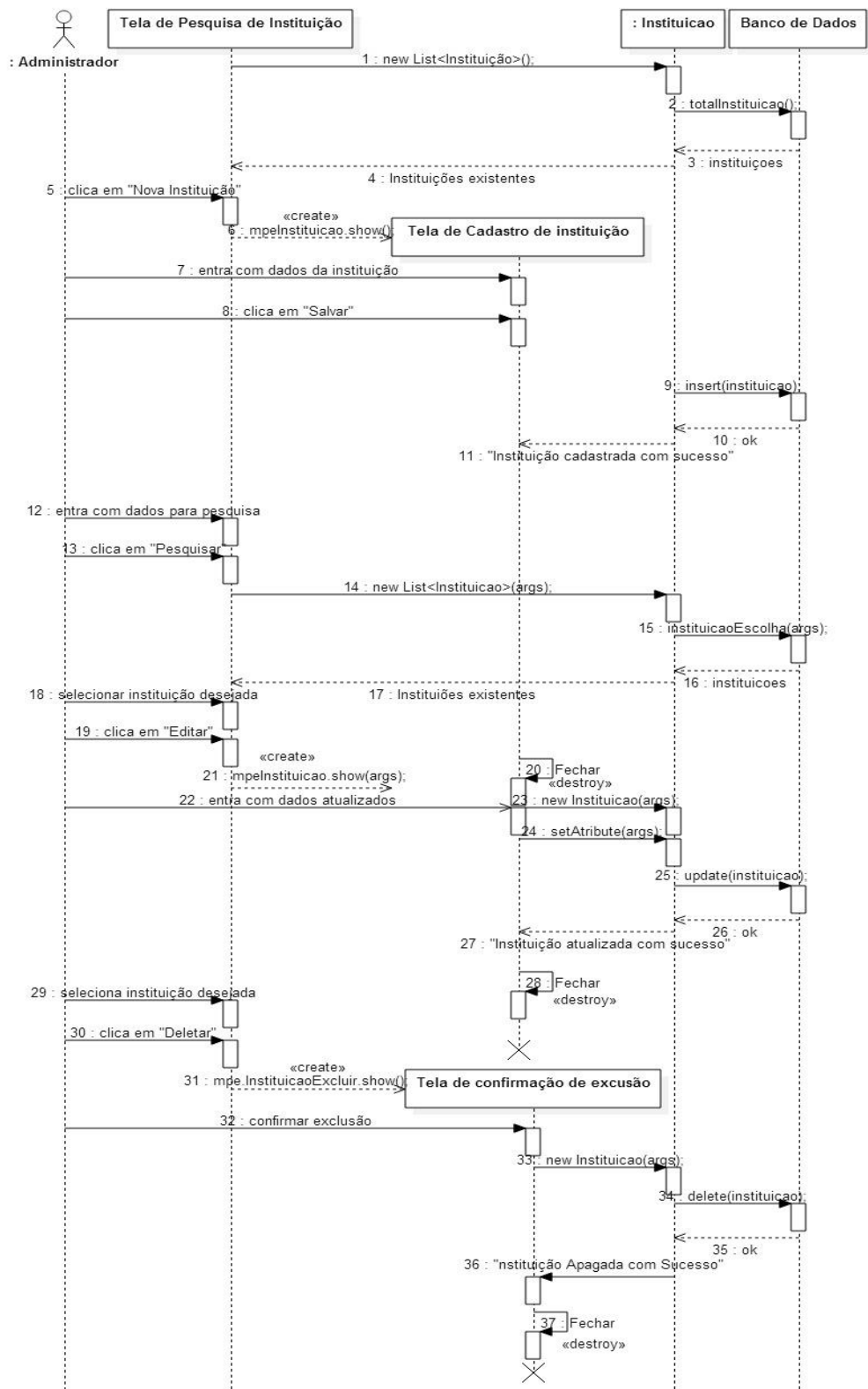




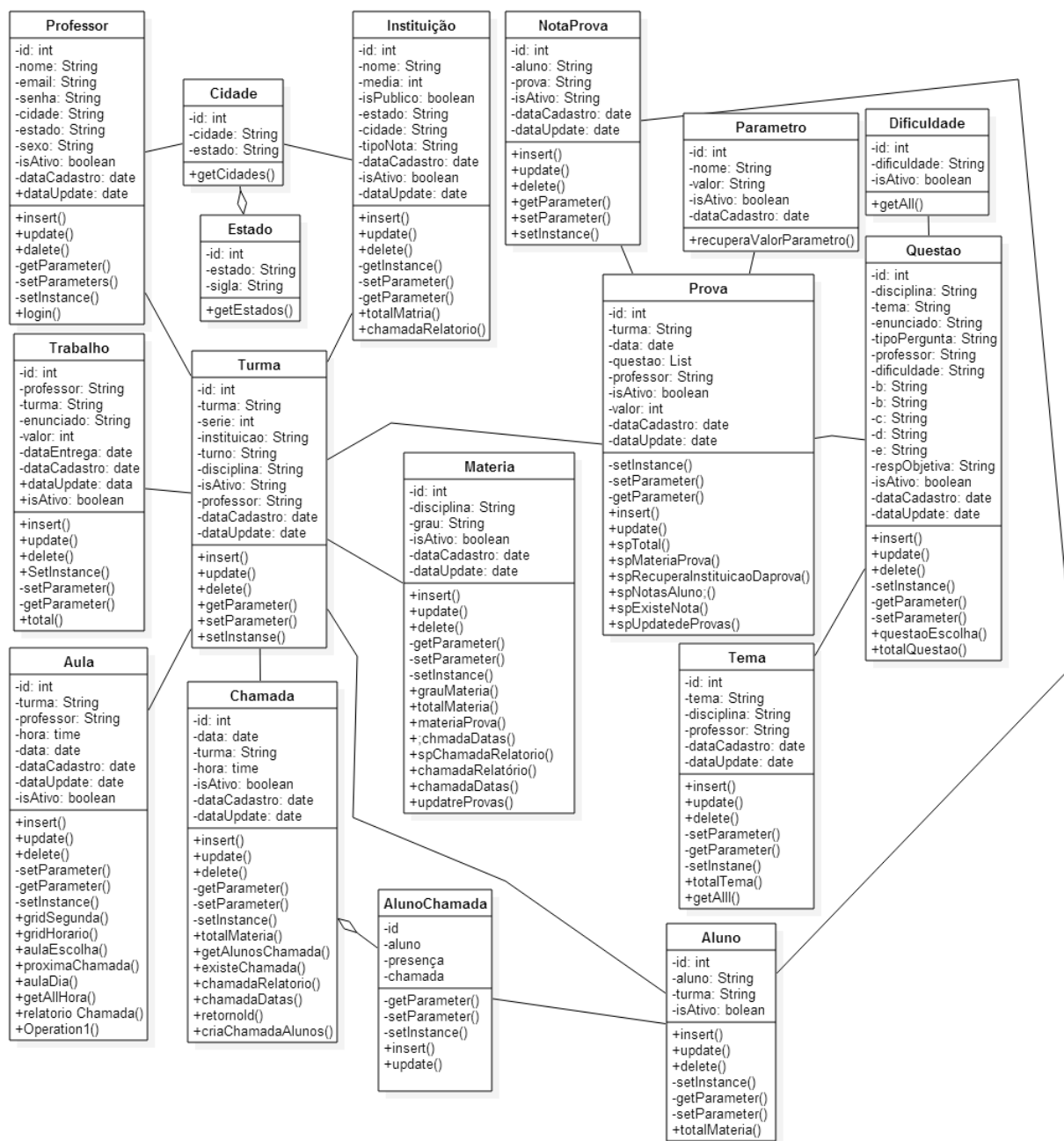
Fazer Chamada:



## Cadastro da instituição de ensino:



## APÊNDICE D – DIAGRAMA DE CLASSES



APÊNDICE E- DIAGRAMA ENTIDADE RELACIONAMENTO (DER)

